

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Regulace vybrané soustavy založená na technologii IQRF
Control of chosen system based on IQRF technology

2018

Marián Antoszyk

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Zadání bakalářské práce

Student: **Marián Antoszyk**
Studijní program: B2649 Elektrotechnika
Studijní obor: 2612R041 Řídicí a informační systémy
Téma: **Regulace vybrané soustavy založená na technologii IQRF**
Control of Chosen System based on IQRF Technology
Jazyk vypracování: čeština

Zásady pro vypracování:

1. Seznámení se s IQRF technologií a DPA protokolem.
2. Seznámení se s řídicím systémem REX na platformě Raspberry PI.
3. Návrh a realizace elektroniky laboratorního modelu pro identifikaci soustav s IQRF transceivery.
4. Identifikace modelu soustavy sestávající z kondenzátorů a rezistorů.
5. Syntéza regulátoru pro identifikovanou soustavu a jeho implementace do řídicího systému REX.
6. Tvorba vizualizace.
7. Zhodnocení výsledků a závěr.

Seznam doporučené odborné literatury:

- [1] NOSKIEVIČ, Petr. *Modelování a identifikace systémů*. Ostrava: Montanex, 1999. 276 s. ISBN 80-722-5030-2.
- [2] VOJTÍŠEK, Rudolf. *Realizace vzdáleně řízené úlohy určené pro virtuální laboratoř založené na moderních algoritmech řízení a technologii HTML5*. Ostrava, 2013. Diplomová práce. Vysoká škola báňská - Technická univerzita Ostrava, Fakulta elektrotechnika a informatiky, Katedra kybernetiky a biomedicínského inženýrství.
- [3] MICRORISC. *IQRF Quick Start Guide For IQRF OS v3.07D and higher*. 2017. [online] Dostupné z: <http://www.iqrf.org>.
- [4] MICRORISC. *IQRF OS Operating System Version 4.00D for (DC)TR-7xD - User's Guide*. 2017. [online] Dostupné z: <http://www.iqrf.org>.
- [5] MICRORISC. *IQRF DPA Framework Technical Guide v3.00*. 2017. [online]. Dostupné z: <http://www.iqrf.org>.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

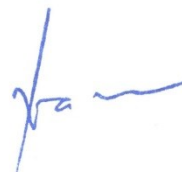
Vedoucí bakalářské práce: **Ing. Martin Pieš, Ph.D.**

Datum zadání: 01.09.2017

Datum odevzdání: 30.04.2018



doc. Ing. Jiří Koziolek, Ph.D.
vedoucí katedry




prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: 26. dubna 2018


.....
podpis studenta

Poděkování

Rád bych poděkoval svému vedoucímu práce, panu Ing. Martinovi Piešovi, Ph.D., za odbornou pomoc a konzultaci při vytváření této bakalářské práce.

Abstrakt

Tato bakalářská práce se zabývá možnostmi regulace soustav s využitím bezdrátové technologie pro přenos hodnot signálových veličin. Zároveň je zde rozebrán a implementován princip regulace pomocí událostmi řízeného regulátoru.

Realizovaná práce využívá technologii IQRF pro bezdrátový přenos, jako řídicí systém je použit REX Controls, jehož jádro běží na platformě Raspberry Pi 3. Vizualizace systému je provozována přes internetové rozhraní využívající technologie HTML5 a JavaScript.

V bakalářské práci jsou teoreticky popsány využití technologie, v praktické části je rozebrán postup návrhu a realizace elektroniky, komunikace, postup identifikace regulované soustavy, návrh regulátoru a tvorba vizualizace.

Realizovaný systém je umístěn na prezentačním panelu a slouží jako demonstrační ukázka propojení bezdrátové komunikace a regulační techniky. Dále bude sloužit jako výukový prostředek na Katedře kybernetiky a biomedicínského inženýrství.

Klíčová slova

IQRF, REX, Raspberry Pi, bezdrátová komunikace, regulace, událostmi řízený regulátor, PID regulátor, identifikace, vizualizace

Abstract

This bachelor's thesis deals with possibilities of process control with the use of wireless technology for the transmission of signal values. The thesis also analyzes the control principle by Event Based Controller.

Realization uses IQRf Technology for wireless transmission, as a control system is used RexControls, which core is running on Raspberry Pi 3 platform. Visualization of the whole system is running via web interface with the use of HTML5 and JavaScript technology.

Theoretical part of bachelor's thesis describes used technologies, practical part analyzes the design process and implementation of hardware components, communication settings, identification process, design of the PID controller and creation of visualization.

Realized system was mounted on a presentation stand and it serves as a demonstration of interconnection two different specializations, specifically, wireless technologies and control systems. Hereafter the stand will serve as an educational resource on Department of Cybernetics and Biomedical Engineering.

Key words

IQRf, REX, Raspberry Pi, wireless communication, regulation, Event Based Control, PID controller, identification, visualization

Obsah

Seznam použitých zkratk	- 9 -
Seznam obrázků	- 10 -
Seznam tabulek	- 11 -
Úvod	- 12 -
1 Technologie bezdrátové komunikace IQRf	- 13 -
1.1 Vývojové sady pro IQRf transceivery	- 14 -
1.2 IQRf komunikace	- 15 -
1.3 Programování IQRf transceiverů	- 16 -
1.3.1 DPA protokol	- 16 -
1.3.2 Custom DPA Handler	- 16 -
1.3.3 Struktura DPA protokolu	- 17 -
1.3.4 IQRf IDE	- 17 -
2 Řídicí systém REX na platformě Raspberry Pi	- 18 -
2.1 Části řídicího systému REX	- 18 -
2.2 Platforma Raspberry Pi	- 20 -
3 Návrh a realizace elektroniky s IQRf transceivery	- 21 -
3.1 Návrh a realizace modulů	- 23 -
3.1.1 Modul DAC-EVAL	- 24 -
3.1.2 Modul ADC-EVAL	- 25 -
3.2 Návrh komunikace	- 26 -
4 Identifikace soustavy	- 29 -
4.1 Analytická identifikace	- 29 -
4.2 Experimentální identifikace	- 30 -
5 Syntéza regulátoru	- 32 -
5.1 Dotazovací algoritmus	- 35 -
6 Vizualizace	- 38 -
7 Zhodnocení výsledků	- 41 -
Závěr	- 43 -
Použitá literatura	- 44 -
Seznam příloh	- 45 -

Seznam použitých zkratk

Zkratka	Význam
CDC	Communications Device Class - Kombinovaná třída USB zařízení
DPA	Direct Peripheral Access - protokol využívaný v IQMESH sítích
DPS	Deska plošných spojů
HMI	Human Machine Interface - Rozhraní mezi zařízením a člověkem
HWP	Hardwarový profil
I2C	Inter-integrated circuit - Synchronní sériová sběrnice
IDE	Integrated Development Environment - Vývojové prostředí
IEC	International Electrotechnical Commission - Mezinárodní elektrotechnická komise
IoT	Internet of Things - Internet věcí
LDO	Low-dropout Regulator - Regulátor napětí
LED	Light-Emitting Diode - Dioda emitující světlo
Li-Pol	Lithium Polymerový akumulátor
MCU	Microcontroller Unit - Jednočipový počítač
OS	Operační systém
Pásmo ISM	Pásmo pro radiové vysílání v průmyslových, vědeckých a zdravotnických oborech
PLC	Programmable Logic Controller - Programovatelný logický automat
RAM	Random Access Memory - Paměť pro čtení a zápis s přímým přístupem
RF	Radio Frequency - Radiová frekvence
SIM	Subscriber Identity Module - Identifikační karta v mobilní síti
SMT	Surface Mount Technology - Technologie povrchové montáže součástek
SPI	Serial Peripheral Interface - Synchronní sériová sběrnice
SSID	Service Set Identifier - Identifikátor bezdrátové sítě Wi-Fi
UART	Universal Asynchronous Receiver-Transceiver - Asynchronní sériová sběrnice
USB	Universal Serial Bus - Univerzální sériová sběrnice

Seznam obrázků

<i>Obr. 1 - Transceiver série TR-72D [2]</i>	- 13 -
<i>Obr. 2 - Proudová spotřeba transceiveru TR-76D [3]</i>	- 14 -
<i>Obr. 3 - DK-EVAL-04 [2]</i>	- 14 -
<i>Obr. 4 - Principiální schéma IQMESH sítě</i>	- 15 -
<i>Obr. 5 - Dva přístupy programování [2]</i>	- 16 -
<i>Obr. 6 - Příklad DPA paketu pro rozsvícení LED [4]</i>	- 17 -
<i>Obr. 7 - Blokové schéma struktury řídicího systému REX [5]</i>	- 18 -
<i>Obr. 8 - RexDraw</i>	- 19 -
<i>Obr. 9 - RexComp</i>	- 19 -
<i>Obr. 10 - RexCore</i>	- 19 -
<i>Obr. 11 - RexView</i>	- 19 -
<i>Obr. 12 - Popis počítače Raspberry Pi 3 - model B</i>	- 20 -
<i>Obr. 13 - Schéma řídicího členu</i>	- 22 -
<i>Obr. 14 - Schéma akčního členu</i>	- 22 -
<i>Obr. 15 - Schéma zpětnovazebního členu</i>	- 23 -
<i>Obr. 16 - Schéma zapojení modulu DAC-EVAL</i>	- 24 -
<i>Obr. 17 - Výsledná podoba modulu DAC-EVAL</i>	- 24 -
<i>Obr. 18 - Schéma zapojení modulu ADC-EVAL</i>	- 25 -
<i>Obr. 19 - Výsledná podoba modulu ADC-EVAL</i>	- 25 -
<i>Obr. 20 - Schéma vytvořené mesh sítě</i>	- 26 -
<i>Obr. 21 - Schéma zapojení dvojitého RC člunku</i>	- 29 -
<i>Obr. 22 - Přejchodová charakteristika analyticky a experimentálně identifikované soustavy</i>	- 31 -
<i>Obr. 23 - Srovnání přechodové charakteristiky soustavy bez regulátoru a s regulátorem</i>	- 32 -
<i>Obr. 24 - Schéma regulačního algoritmu v programu RexDraw</i>	- 34 -
<i>Obr. 25 - Znázornění stavů v řídicím algoritmu a systém dotazování</i>	- 36 -
<i>Obr. 26 - Vizualizace regulačního obvodu</i>	- 38 -
<i>Obr. 27 - Vizualizace záznamu komunikace</i>	- 39 -
<i>Obr. 28 - Vizualizace nastavení regulátoru</i>	- 40 -
<i>Obr. 29 - Vytvořený prezentační stojan</i>	- 41 -

Seznam tabulek

<i>Tabulka 1: DPA požadavek pro zápis dekadické hodnoty na D/A převodník (hex. tvar)</i>	<i>- 26 -</i>
<i>Tabulka 2: DPA odpověď zápisu dekadické hodnoty na D/A převodník (hex. tvar).....</i>	<i>- 27 -</i>
<i>Tabulka 3: DPA požadavek pro získání aktuální hodnoty napětí na daném kanálu (hex. tvar)</i>	<i>- 27 -</i>
<i>Tabulka 4: DPA odpověď aktuální hodnoty napětí na daném kanálu (hex. tvar).....</i>	<i>- 27 -</i>
<i>Tabulka 5: DPA požadavek pro získání aktuální hodnoty napětí na výstupu LDO regulátoru u uzlu s modulem ADC-EVAL (hex. tvar).....</i>	<i>- 28 -</i>
<i>Tabulka 6: DPA odpověď aktuální hodnoty napětí výstupu LDO regulátoru u uzlu s modulem ADC-EVAL (hex. tvar).....</i>	<i>- 28 -</i>
<i>Tabulka 7: DPA požadavek pro získání aktuální hodnoty napětí na výstupu LDO regulátoru u uzlu s modulem DAC-EVAL (hex. tvar).....</i>	<i>- 28 -</i>
<i>Tabulka 8: DPA odpověď aktuální hodnoty napětí výstupu LDO regulátoru u uzlu s modulem DAC-EVAL (hex. tvar).....</i>	<i>- 28 -</i>
<i>Tabulka 9: Srovnání vlastností odezev na jednotkový skok</i>	<i>- 33 -</i>

Úvod

Cílem této práce je vytvořit regulační úlohu, kde přenos hodnot mezi jednotlivými částmi regulačního obvodu bude realizován bezdrátově s využitím IQRF technologie. U fyzikálních soustav je čas od času nepraktické a náročné instalovat vedení od akčního prvku a zároveň snímacího prvku do řídicí jednotky, která je od procesu vzdálená. Navíc toto vedení bývá pouze signálové a tím pádem ho lze nahradit právě radiofrekvenčním signálem. Mimo jiné existují fyzikální soustavy vyžadující regulační zásah pouze v krátkém časovém intervalu, naopak časový interval mezi dvěma událostmi je dlouhý, neboť se nemění požadovaná vstupní veličina. Tím pádem je v této oblasti prostor pro návrh a implementaci sofistikovaných algoritmů, které svým zasíláním dotazů co nejméně zatěžují přijímače napájené pouze z baterie, a tím lze zvýšit jejich výdrž. Takovému přístupu regulace se rovněž říká Event Based Control.

Obsahem prvních dvou kapitol je teoretické seznámení s technologiemi, které jsou použity k realizaci výsledného systému. Konkrétně se jedná o bezdrátovou technologii IQRF, řídicí systém REX a platformu Raspberry Pi 3. Třetí kapitola se věnuje struktuře návrhu regulačního systému a realizaci hardwarové části potřebné k řešení práce. Zároveň je zde detailně popsána navržená bezdrátová komunikace pomocí technologie IQRF. Ve čtvrté kapitole je popsána metoda analytické a experimentální identifikace regulované soustavy. Na toto téma plynule navazuje pátá kapitola zabývající se návrhem konstant regulátoru a řešením vedoucím k vytvoření událostmi řízeného regulátoru. Další kapitola shrnuje tvorbu vizualizace běžící na webovém prohlížeči, v poslední kapitole je zhodnocení dosažených výsledků celé práce.

1 Technologie bezdrátové komunikace IQRF

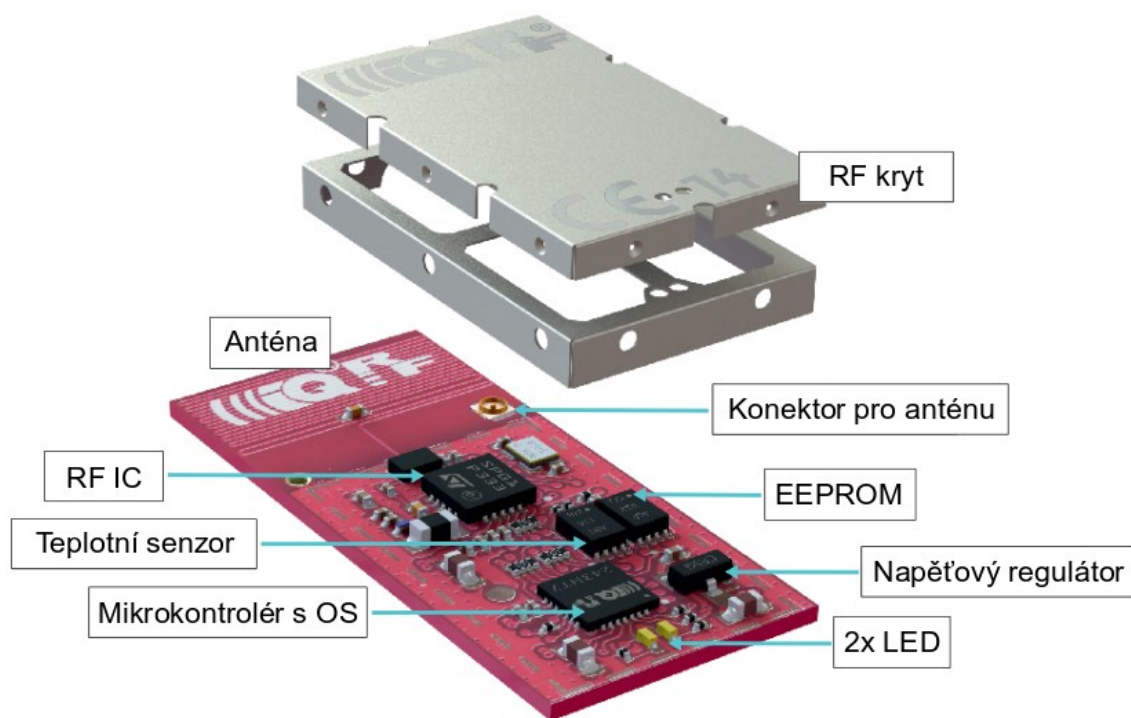
IQRF je komplexní technologické řešení, které umožňuje inteligentní bezdrátové připojení. Zahrnuje hardware, software, komunikační protokoly, dále také podporu, služby či IQRF cloud. Technologie byla vyvinuta českou firmou MICRORISC a poprvé představena v roce 2004 na konferenci ve španělské Malaze. Od roku 2017 je nově součástí odštěpené firmy IQRF Tech s.r.o. [1]

Technologie IQRF je určena především pro řešení v oblastech IoT a machine to machine komunikaci. Lze ji použít téměř u jakéhokoliv elektrického zařízení. Vyznačuje se zejména velmi nízkou spotřebou a dosahem signálu až stovek metrů, v některých případech lze v otevřeném prostoru dosáhnout až jednotek kilometrů. Síť využívá bezlicenční frekvenční pásma ISM 868 MHz pro Evropu, 916 MHz pro USA a 433 MHz celosvětově. Data se přenáší v paketech o maximální velikosti 64 B, rychlost přenosu je maximálně 19,8 kb/s.

Mezi další výhody technologie se řadí:

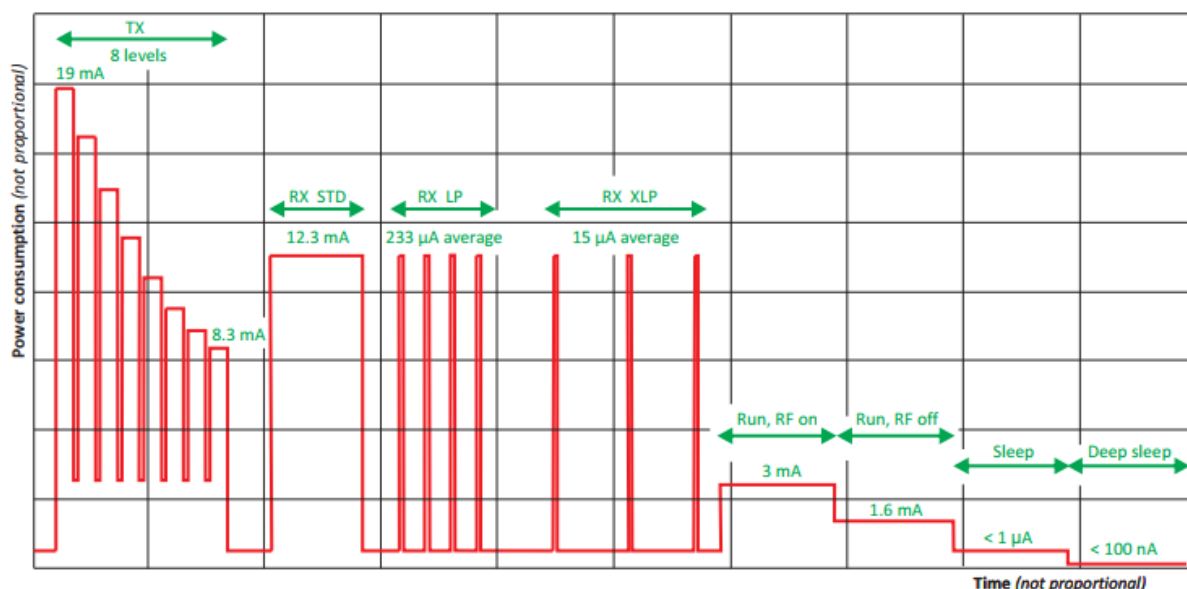
- Jednoduchá a rychlá implementace
- Spolehlivost komunikace
- Integrovaný vlastní IQRF OS umožňující aplikace programované v jazyce C
- DPA komunikační vrstva
- Možnost vytváření IQMESH sítí

Základní komponentou technologie jsou tzv. IQRF transceivery. Jedná se o komponentu velikosti SIM karty, která obsahuje kompletní implementaci pro bezdrátovou konektivitu. Jednotlivé části zařízení jsou popsány v následujícím obrázku. [2]



Obr. 1 - Transceiver série TR-72D [2]

IQRF transceivery mají své číselné označení a odlišují se v různých hlediscích. Dnes již zastaralá řada se označuje TR-5xD Family a je modré barvy, novější řada TR-7xD Family má barvu červenou. Od modelů TR-7xD a výše mají IQRF transceivery nižší proudovou spotřebu než v přechozích verzích. Dle zvoleného módu mohou mít při příjmu odběr nejméně 16,3 μA (Extra Low Power Mode), v Extra Sleep módu dokonce až 56 nA. Velikost proudového odběru při různých stavech zařízení je znázorněna v grafu. Připevnit k elektronickému zařízení se dají SMT montáží, zasunutím do SIM konektoru nebo RJ-45 konektoru. [2]



Obr. 2 - Proudová spotřeba transceiveru TR-76D [3]

1.1 Vývojové sady pro IQRF transceivery

Pro nahrání vlastní aplikace a konfigurace je nutno mít k dispozici HW programátor/debugger CK-USB-04. Do něj se vloží IQRF transceiver a pomocí micro USB konektoru se modul připojí k počítači.

Pro provoz IQRF transceiverů se využívá tzv. Node kitů. Ty mají výrobní označení DK-EVAL-04. Transceiver se vloží do modulu a celý modul se umístí do cílového zařízení. Modul není nutno mít stabilně připojen k napájení, neboť je napájen z Li-Pol akumulátoru s výdrží 400 mAh.

K jednotlivým uzlům je poté možné připojit další moduly. Výhodou je, že se moduly dají kaskádně řetězit. Samotná firma IQRF Tech s.r.o. nabízí oficiální přídatné periferní moduly, například DDC-IO-01 pro I/O řízení, DDC-SE-01 obsahující teplotní senzor, fotorezistor a potenciometr nebo DDC-RE-01 se dvěma relé. Moduly lze propojit pomocí sběrnic SPI, I2C případně UART. [2]

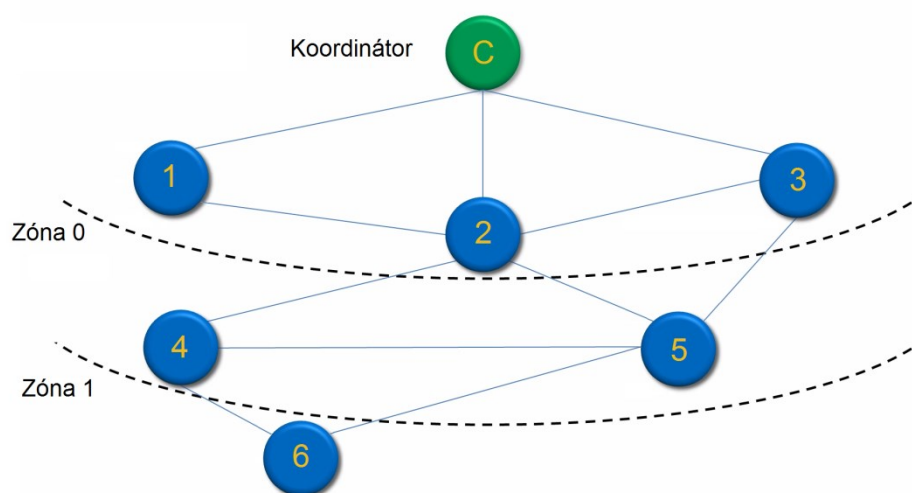


Obr. 3 - DK-EVAL-04 [2]

1.2 IQRF komunikace

IQRF technologie využívá dvou typů komunikací. Jednodušší variantou je komunikace typu peer-to-peer. Všechna zařízení v síti jsou na stejné úrovni a jsou mezi sebou spojeny přímou vazbou. Při vyslání zprávy jedním zařízením je zpráva obdržena všemi ostatními zařízeními v dosahu. Zpracování zprávy poté závisí zcela na programátorovi, který musí v jednotlivých zařízeních definovat, jak s přijatými daty dále pracovat. Tento typ komunikace je využíván zejména při spojení dvou zařízení.

Druhou možností komunikace je vytvoření mesh sítě. IQRF tuto technologii označuje pod názvem IQMESH. Princip této komunikace spočívá ve směrování datových paketů přes ostatní zařízení (tzv. uzly) v síti. Koordinátor, který řídí celou síť, vysílá paket všem uzlům, které má v dosahu. Uzly po obdržení paketu podle adresy rozeznají, zda je zpráva pro ně či ne. Pokud zpráva není určena pro ně, přeposílají ji dalším uzlům v síti, které jsou v jejich dosahu. Tím lze dosáhnout komunikace na větší vzdálenosti a výhodou je zejména robustnost komunikace, neboť se vzdálenými uzly lze komunikovat i přes výpadek některých uzlů nebo komunikačních linek v mesh topologii.



Obr. 4 - Principiální schéma IQMESH sítě

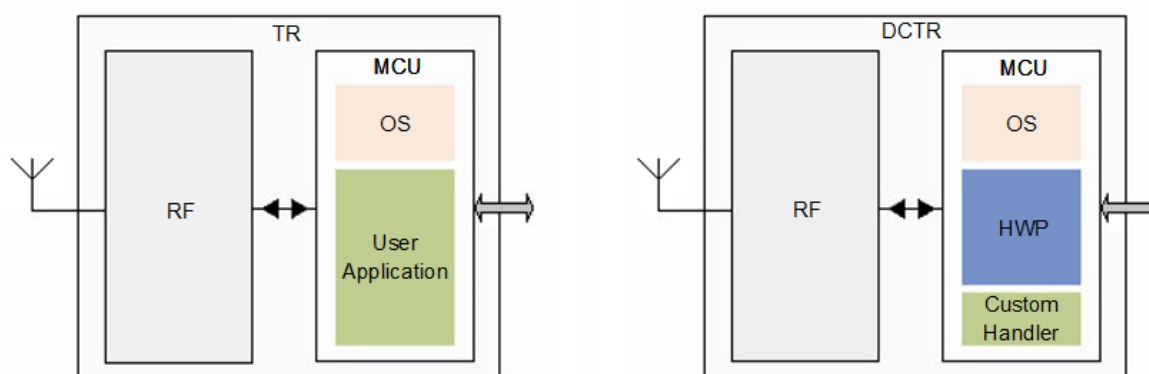
V jedné IQMESH síti může komunikovat až 240 zařízení. Každý transceiver může pracovat jako koordinátor nebo uzel, záleží jen na jeho nastavení. Chceme-li přidat uzly do mesh sítě, musíme každému z nich přiřadit unikátní adresu pro jednoznačnou identifikaci. Přiřazení adresy se vyvolává softwarově a následně je nutné fyzicky stisknout tlačítko na příslušném uzlu. Pro sestavení topologie sítě IQRF implementuje funkci Discovery, která automaticky v reálném čase vyhledá veškerá zařízení a cesty k nim. Komunikace mezi zařízeními je v IQRF technologii zajištěna operačním systémem, ve kterém je integrovaný framework DPA. [2]

1.3 Programování IQRF transceiverů

K programování můžeme přistupovat dvěma způsoby. Buď lze vytvořit kompletní uživatelskou aplikaci v jazyce C, nebo lze nahrát hardwarový profil a DPA Handler.

U prvně zmíněného přístupu je výhodou větší flexibilita a efektivnost programu, nevýhodou však je, že pomocí tohoto přístupu není doporučeno vytvářet IQMESH síť, neboť tato činnost vyžaduje mnoho úsilí. Další nevýhodou je také větší časová náročnost při vývoji, a proto se hodí spíše pro rozsáhlejší aplikace.

Druhý přístup využívá komunikační protokol nazvaný DPA, čímž odpadá nutnost programování a operace jsou prováděny jednoduchými příkazy. Využití tohoto přístupu vede k jednoduššímu a rychlejšímu vývoji výsledné aplikace a lze pomocí něj vytvořit IQMESH síť.



Obr. 5 - Dva přístupy programování [2]

1.3.1 DPA protokol

DPA je jednoduchý bytově orientovaný protokol, který funguje jako komunikační vrstva nad OS. IQRF transceiver je vybaven hardwarovým profilem (HWP), což je softwarový plugin zajišťující DPA protokol. Místo programování jsou specifické funkce dosaženy pouze jednoduchými příkazy přes datový tok.

Komunikace probíhá tak, že koordinátor zašle požadavek na danou činnost (vyčtení teploty z periferie, zapnutí LED...) a následně nadřazenému systému zašle potvrzovací zprávu s parametry komunikace. Pokud byl požadavek správný a nedošlo k výpadku při komunikaci, po určité době koordinátor obdrží odpověď, kterou opět předá nadřazenému systému. Doba odezvy je závislá na počtu skoků mezi uzly, typu nahraného hardwarového profilu v IQRF transceiveru, typu požadavku a typu dotazování. [4]

1.3.2 Custom DPA Handler

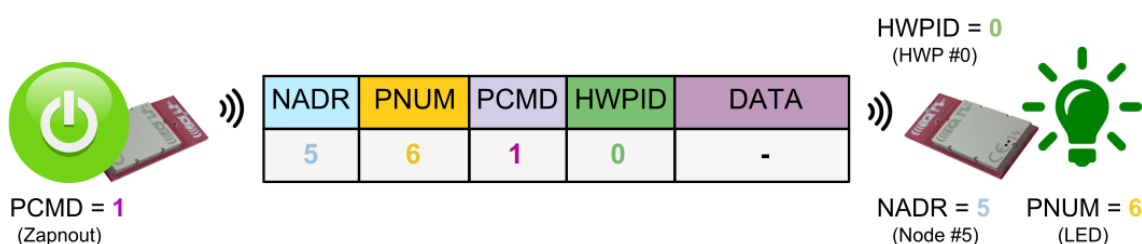
Standardní externí periferie koncových zařízení (Node kitů) mohou být obsluhovány pomocí standardních příkazů DPA. Je-li potřeba obsluhovat uživatelsky vytvořené periferie, případně potřebujeme-li rozšířit funkcionalitu chování externího zařízení, je nutno použít Custom DPA Handler. Jedná se o kód naprogramovaný v jazyce C. Pokud je Custom DPA Handler implementován do zařízení, musí být také povolen v HWP konfiguraci. [4]

1.3.3 Struktura DPA protokolu

Každý vyslaný paket přes protokol DPA se skládá ze čtyř, případně pěti částí. Tyto části v sobě nesou následující informace:

- NADR (Network address) – číselné označení modulu v síti
- PNUM (Peripheral number) – číselné označení periferie daného modulu
- PCMD (Peripheral command) – příkaz pro danou periferii
- HWPID (HWP identification) – identifikátor hardwarového profilu
- DATA – datová zpráva pro daný modul, je nepovinná a uživatelsky volitelná

Následující obrázek uvádí příklad zprávy, kterou koordinátor vyšle pomocí DPA protokolu, pro zapnutí LED na uzlu. [4]



Obr. 6 - Příklad DPA paketu pro rozsvícení LED [4]

1.3.4 IQRF IDE

Pro zjednodušení práce při vytváření SW části nabízí IQRF technologie také své vlastní vývojové prostředí. Lze si vybrat mezi verzí s uživatelským rozhraním a verzí komunikující přes příkazový řádek. Druhá zmíněná verze je určena spíše pro zkušené vývojáře, případně pro nezaučenou obsluhu, která může jedním příkazem spouštět dávkové soubory a vyhnout se tak chybám. IQRF IDE nabízí veškeré prostředky pro práci s transceivery jako např. prostředky pro nahrávání souborů do transceiverů, debugger, komunikační terminál, IQMESH manažer či nástroje pro upgrade OS. Instalace obsahuje také C kompilátor, který je možné spouštět přímo z IQRF IDE. [4]

2 Řídicí systém REX na platformě Raspberry Pi

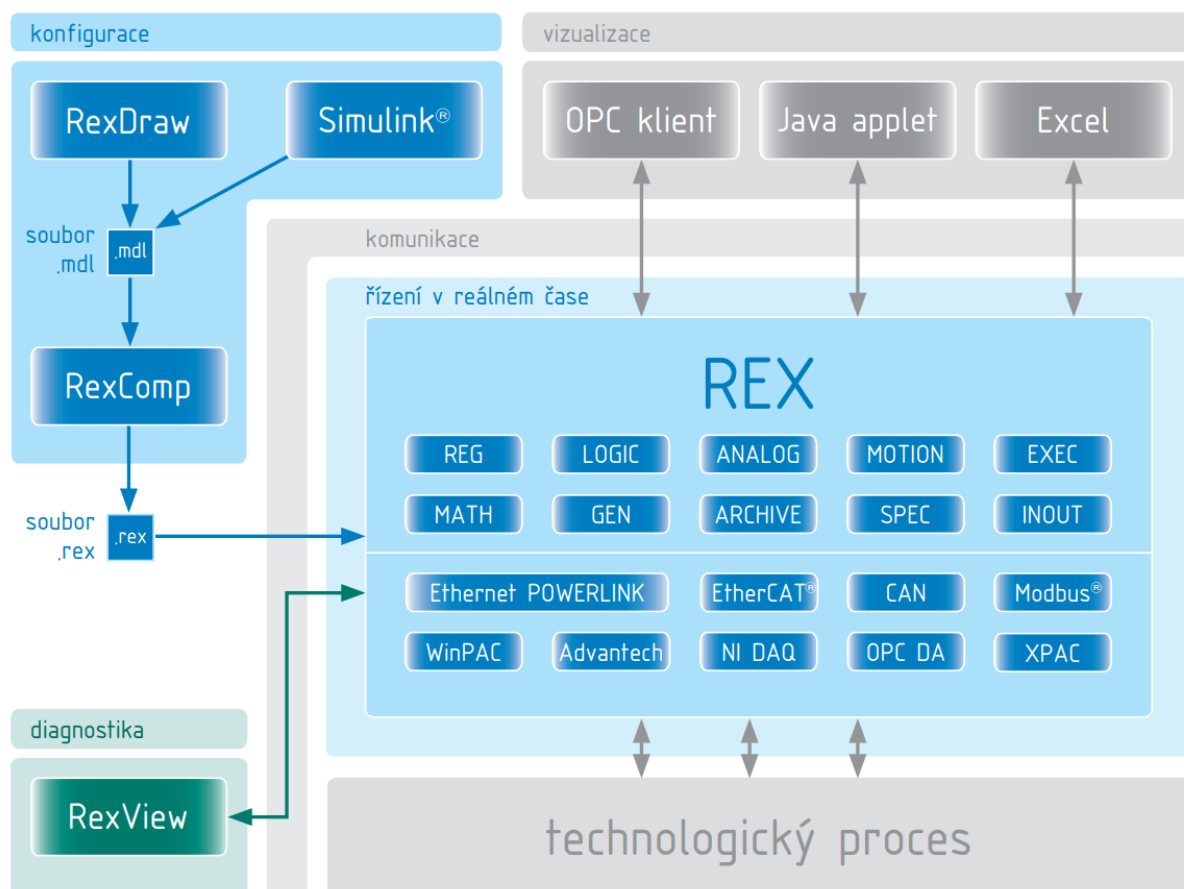
Řídicí systém REX je komplexní softwarový nástroj používaný v oblastech využívajících řízení, jako je automatizace, robotika, měření či regulace. Systém byl vyvinut českou společností REX Controls s.r.o. sídlící v Plzni. Technologie se specializuje zejména pro využití ve vývoji a výzkumu a pro nestandartní, prototypová a experimentální zařízení.

Řídicí systém REX lze provozovat na různých operačních systémech jako například Windows XP a výše, Windows Embedded, GNU/Linux či PharLap ETS. Velkou výhodou systému REX je kompatibilita se světově využívaným produktem Matlab/Simulink, ve kterém lze simulovat a testovat navržené algoritmy. [5]

2.1 Části řídicího systému REX

Princip činnosti řídicího systému REX je přehledně znázorněn v blokovém schématu (Obr. 7). Systém se skládá z několika dílčích částí:

- RexDraw - Grafické vývojové prostředí
- RexComp - Kompilátor
- RexCore - Runtime jádro
- RexView - Diagnostika runtime jádra

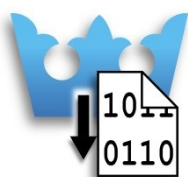


Obr. 7 - Blokové schéma struktury řídicího systému REX [5]

RexDraw je vývojové prostředí řídicích algoritmů založené na grafickém programování pomocí funkčních bloků (FBD) dle platné mezinárodní normy IEC 61131-3, které se využívá zejména při programování PLC. K dispozici jsou rozsáhlé knihovny, ve kterých lze najít funkční bloky sloužící například pro práci s analogovými signály, generátory signálů, logické operace, matematické operace, čítače a časovače, PID regulátory atd. Výsledkem programování je soubor s příponou *.mdl*. Při běhu naprogramovaného řídicího algoritmu je také v prostředí možno monitorovat signály a funkční bloky. [5]



Obr. 8 - RexDraw



Obr. 9 - RexComp

RexComp slouží jako překladač, který převádí algoritmy vytvořené ve vývojovém prostředí RexDraw, případně v prostředí Matlab/Simulink, s příponou *.mdl* do binárního kódu systému REX. Kompilátor je spouštěn přímo z prostředí RexDraw. Při překladač jsou vypisovány informace o vybraném souboru a případně vzniklých chybách. Výstupem kompilace je soubor, který má příponu *.rex*. [5]

RexCore je nejdůležitější částí celého řídicího systému REX. Tento software běží přímo v cílovém zařízení a jedná se o proces, který vykonává naprogramovaný řídicí algoritmus v reálném čase. Program se dle časování cyklicky opakuje a vykonává jednotlivé úlohy v režimu preemptivního multitaskingu, tedy o přidělování a odebírání procesorového času se stará operační systém na základě priorit jednotlivých úloh. Běh runtime jádra je licencován, v demoverzi může běžet maximálně 2 hodiny. [5]



Obr. 10 - RexCore



Obr. 11 - RexView

RexView se využívá pro sledování běhu runtime jádra RexCore. Pomocí něj lze diagnostikovat běh řídicího algoritmu v reálném čase a provádět úpravu jeho parametrů. Program poskytuje detailní informace o všech činnostech jádra RexCore. Tento nástroj nachází uplatnění zejména při ladění a uvádění řídicích systémů do provozu. Připojení k cílovému zařízení je možné i vzdáleně přes TCP/IP protokol. [5]

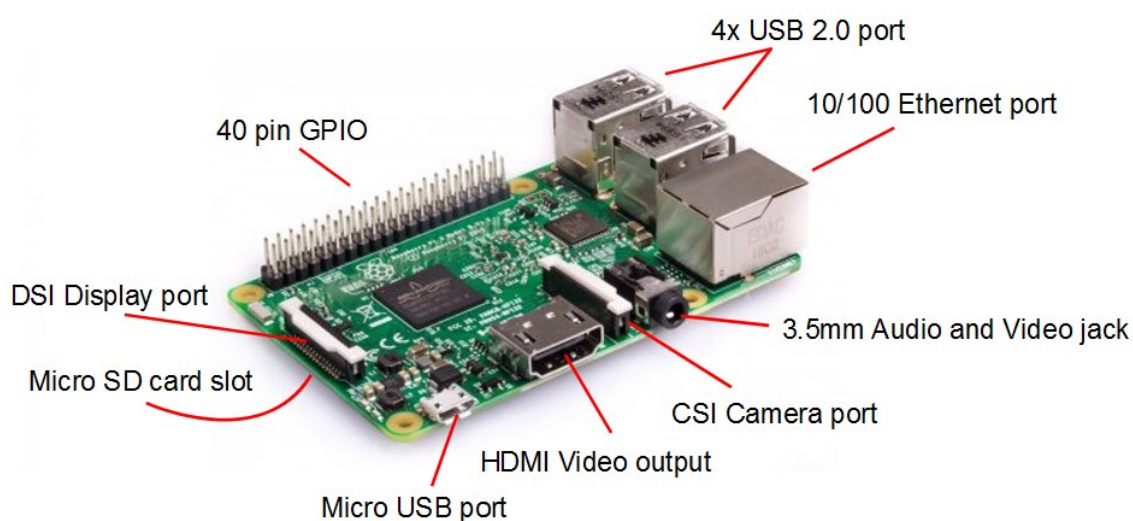
Mezi další část lze uvést také **RexHMI**. Jedná se o program, pomocí něhož lze vytvořit uživatelské rozhraní pro řídicí aplikaci. RexHMI využívá značkovací jazyk HTML5, který funguje v běžných internetových prohlížečích pod mnoha operačními systémy. Tím pádem lze vizualizaci spustit na široké škále zařízení (stolní počítače, laptopy, tablety aj.). Vizualizaci můžeme vytvořit těmito třemi metodami:

- WebWatch - Jedná se o automaticky vygenerované HMI přímo z prostředí RexDraw.
- RexHMI Designer - Vizualizaci vytváří uživatel, prostředí vychází z editoru Inkscape.
- WebBuDi - Vizualizaci vytváří uživatel pomocí skriptů v JavaScriptu.

2.2 Platforma Raspberry Pi

Raspberry Pi je malý jednodeskový počítač o velikosti 85,6 mm x 56 mm vyvinutý v roce 2012 britskou nadací Raspberry Pi Foundation. Hlavní myšlenkou tohoto zařízení je přiblížit využití digitální techniky a principy programování lidem po celém světě, nejvíce však tato platforma cílí na studenty. [6]

Raspberry Pi je dostupné v několika variantách, které se odvíjejí zejména podle historického vývoje zařízení. Současně nejnovějším produktem je model třetí generace, který byl na trh uveden v únoru 2016. Oproti svým předchůdcům se nově vyznačuje možností bezdrátové komunikace pomocí WiFi 802.11 a Bluetooth 4.1 LE. Model je vybaven 64bitovým čtyřjádrovým procesorem ARM Cortex-A53 s taktem 1,2 GHz a 1 GB paměti RAM. Zařízení je napájeno z externího zdroje přes Micro USB port. [7]



Obr. 12 - Popis počítače Raspberry Pi 3 - model B

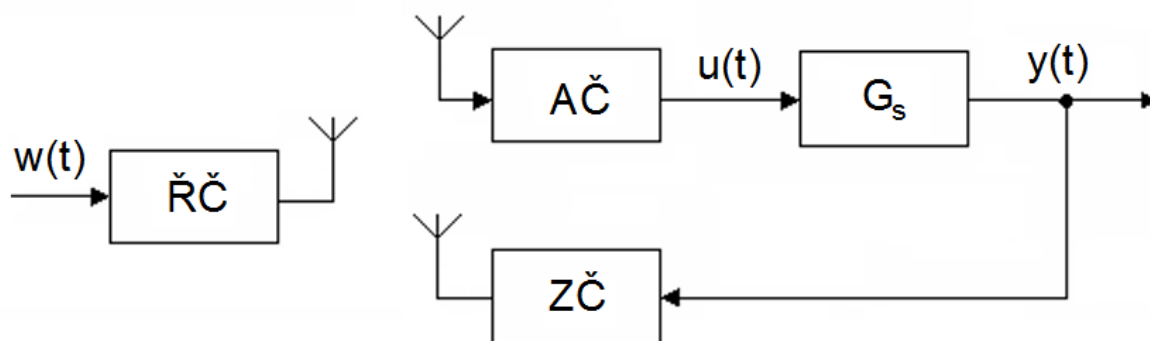
Platforma podporuje širokou škálu operačních systémů, přičemž většinu lze pořídit bezplatně. Samotnou nadací oficiálně vyvíjený operační systém distribuce GNU/Linux se jmenuje Raspbian. Ten je odvozen z operačního systému Debian, který je vyvíjen dobrovolníky a nadšenci z celého světa. Raspberry Pi nedisponuje interním uložištěm, proto je nutné operační systém nahrát na externí uložiště (SD kartu, USB flash disk...) a připojit jej k zařízení.

3 Návrh a realizace elektroniky s IQRF transceivery

Cílem návrhu bylo vytvořit regulační úlohu, kde přenos dat mezi jednotlivými částmi bude realizován bezdrátově s využitím IQRF technologie. Pro zjednodušení popisu lze celý regulační obvod rozdělit na 4 parciální části:

- **Řídicí člen** – Jedná se o hlavní článek regulačního obvodu, který má za úkol vykonávat naprogramovaný regulační algoritmus a také zajišťovat komunikaci mezi jednotlivými částmi obvodu (sloužit jako koordinátor sítě).
- **Akční člen** – Slouží pro příjem hodnot akčního zásahu z řídicího členu, kterým působí na regulovanou soustavu.
- **Regulovaná soustava** – Systém, který je řízen. V této práci je to dvojitý RC článek.
- **Zpětnovazební člen** – Část, která zajišťuje přenos dat z výstupu regulované soustavy zpět do řídicího členu.

Principiální schéma struktury vychází ze základního schématu regulačního obvodu a je zobrazeno v následujícím blokovém schématu.



ŘČ – Řídicí člen

AČ – Akční člen

Gs – Přenos regulované soustavy

ZČ – Zpětnovazební člen

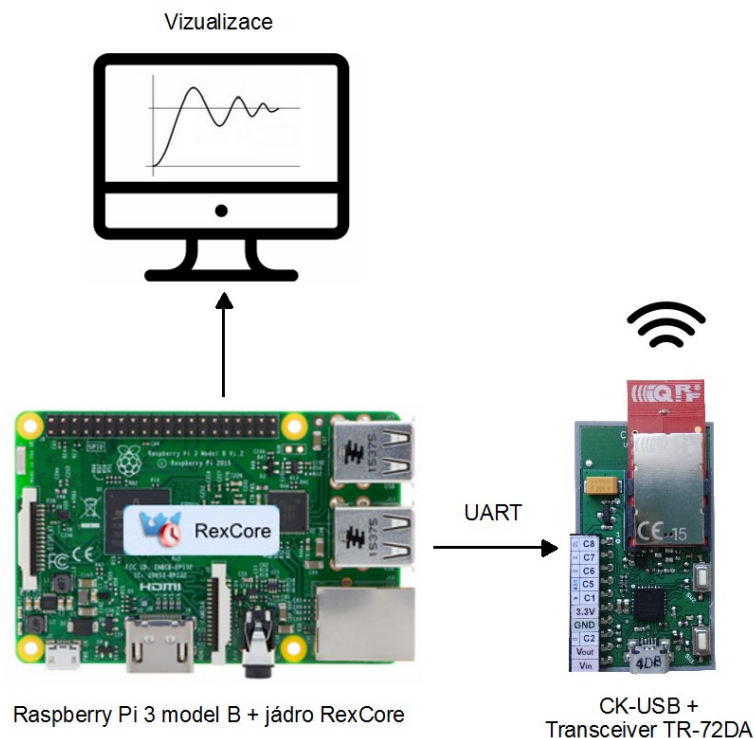
$w(t)$ - Žádaná hodnota

$u(t)$ – Akční zásah

$y(t)$ – Regulovaná veličina

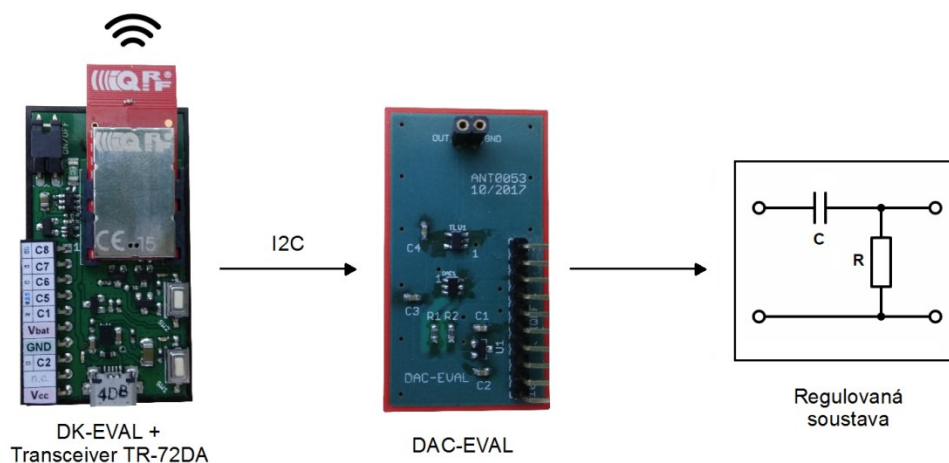
Žádaná hodnota $w(t)$ je zadávána uživatelsky přes vizualizační rozhraní pomocí ovládacího prvku. V řídicím členu probíhá výpočet regulační odchylky dle známého vzorce $e(t) = w(t) - y(t)$ a zároveň tento člen působí jako regulátor, takže vypočítává dle implementovaného regulačního algoritmu okamžité hodnoty akčního zásahu $u(t)$.

Řídicí člen je tvořen jednodeskovým počítačem Raspberry Pi 3, na kterém běží runtime jádro řídicího systému REX, v němž je vytvořen regulační algoritmus. Tím se z platformy stává řídicí jednotka typu SoftPLC. K RPi3 je připojen programátor/debugger CK-USB-04A v 1.00, ve kterém je umístěn IQRF transceiver sloužící pro bezdrátovou komunikaci. Programátor je přepnut do CDC SPI módu, tudíž mikroprocesor s transceiverem komunikuje pomocí emulované sériové sběrnice.



Obr. 13 - Schéma řídicího členu

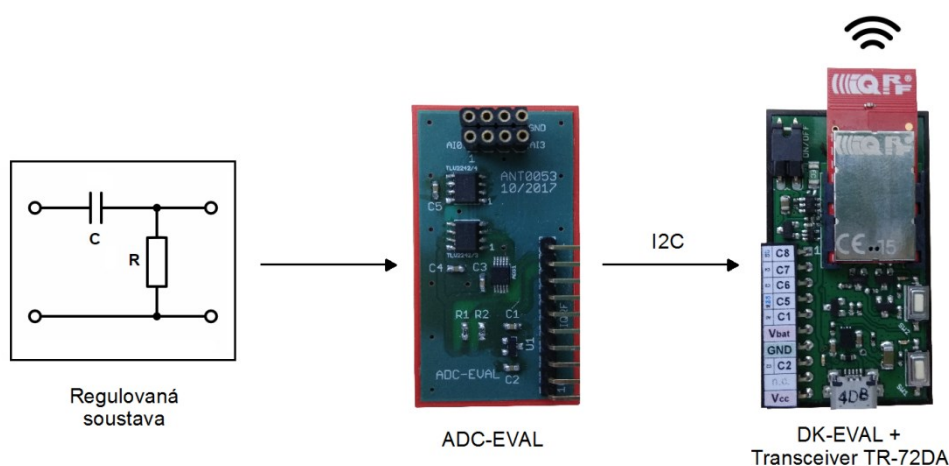
Akční člen se skládá z modulu DK-EVAL-04A v1.01, do kterého je vložen IQRF transceiver. Ten přijímá z řídicího členu hodnoty akčního zásahu v digitální podobě, proto je k němu nutno připojit vytvořený modul DAC-EVAL s D/A převodníkem.



Obr. 14 - Schéma akčního členu

Regulovaná soustava je v této práci realizována jako libovolná kombinace rezistorů a kondenzátorů zapojených na nepájivém poli. Výhodou je, že součástky lze jednoduše obměňovat a vytvářet tak soustavy různých řádů a s různými časovými konstantami. Další výhodou je také kompletní znalost matematicko-fyzikálních vztahů, což umožňuje soustavu identifikovat nejen experimentálně měřením odezvy na jejím výstupu, ale také analyticky sestavením matematického modelu.

Zpětnovazební člen je složen z uživatelsky vytvořeného modulu ADC-EVAL, který slouží pro převod analogové hodnoty napětí na digitální hodnotu. K němu je připojen přes sběrnici I2C modul DK-EVAL-04A v1.01, do něhož je vložen IQRF transceiver.



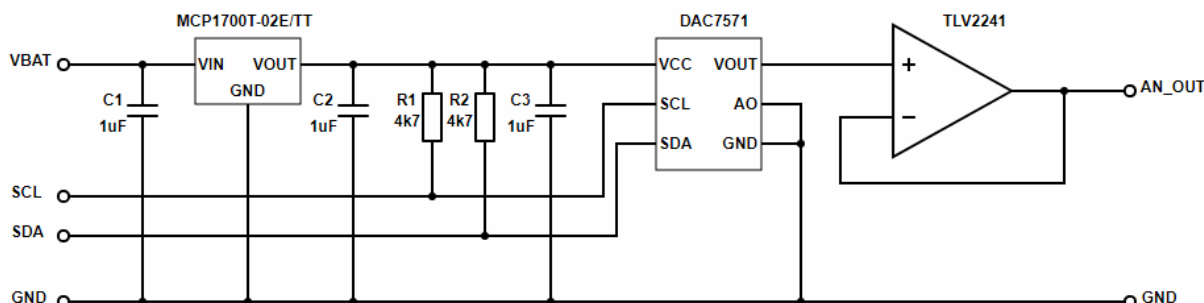
Obr. 15 - Schéma zpětnovazebního členu

3.1 Návrh a realizace modulů

Při návrhu modulů byl kladen důraz především na minimalizaci celého zařízení a výběr součástek s co nejmenším proudovým odběrem. Dalším požadavkem bylo, aby se moduly daly do sebe jednoduše zapojit. Velikost DPS a krabičky byla tedy zvolena stejná, jako jsou velikosti modulů DK-EVAL-04A. Konkrétně deska plošných spojů má velikost 25x47 mm, vnější průměr krabičky je 27x49 mm. Krabička pro osazenou DPS byla vytištěna na 3D tiskárně a zároveň byla barevně odlišena od modulů s periferiemi, které jsou oficiálně nabízeny firmou IQRF Tech s.r.o. Kromě pinů a pinových lišt pro spojení s modulem DK-EVAL-04A byly všechny součástky pro osazení vybrány pro technologii povrchové montáže.

3.1.1 Modul DAC-EVAL

Modul byl vytvořen pro převod digitálních hodnot velikosti akčního zásahu, které zasílá řídicí člen, na analogovou hodnotu napětí, která působí na regulovanou soustavu. Obvodové zapojení znázorňuje níže uvedené schéma.



Obr. 16 - Schéma zapojení modulu DAC-EVAL

Napájecí napětí modulu je přivedeno z baterie na modulu DK-EVAL a dále je řízeno napěťovým regulátorem MCP1700T s proudovým odběrem $1,6 \mu\text{A}$, který udržuje napětí na konstantní úrovni $3,3 \text{ V}$ a zároveň chrání modul proti zkratu a přehřátí.

Jako D/A převodník byl použit typ DAC7571, který se vyznačuje zejména nízkou proudovou spotřebou při převodu ($120 \mu\text{A}$ při napájecím napětí 3 V), komunikací pomocí I2C sběrnice a pro úlohu dostatečným 12bitovým rozlišením, takže při napájecím napětí $3,3 \text{ V}$ je úroveň kvantizační hladiny zaokrouhleně $0,8 \text{ mV}$. Převodník je také možno přepnout do power-down módu, přičemž proudová spotřeba klesne na 50 nA .

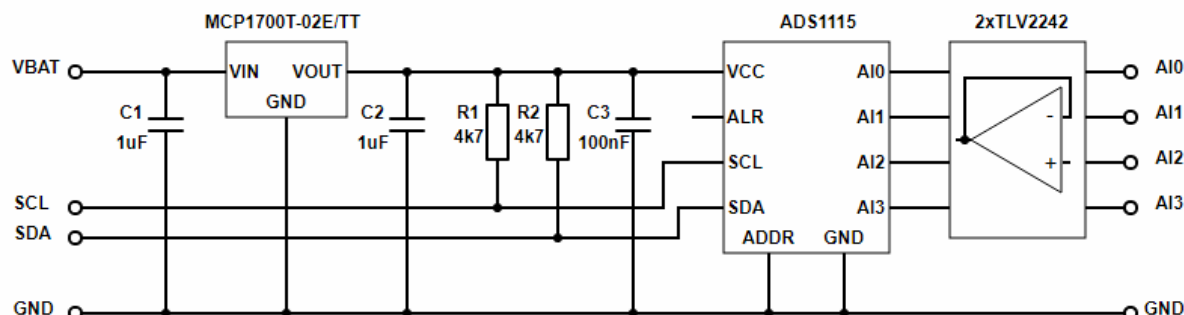
Výstup je impedančně oddělen emitorovým sledovačem, na což byl použit operační zesilovač TLV2241 s nízkým proudovým odběrem a rail-to-rail vstupem a výstupem.



Obr. 17 - Výsledná podoba modulu DAC-EVAL

3.1.2 Modul ADC-EVAL

Modul slouží pro převod analogové hodnoty napětí na výstupu regulačního obvodu na digitální hodnotu, která je poté bezdrátově zaslána do řídicího členu. Obvodové zapojení je vyobrazeno v následujícím schématu.



Obr. 18 - Schéma zapojení modulu ADC-EVAL

Napájecí napětí modulu je stejně jako u předchozího modulu DAC-EVAL řízeno napěťovým regulátorem MCP1700T.

Pro převod analogového signálu na digitální byl zvolen převodník ADS1115 typu sigma-delta. Mezi jeho klíčové parametry patří nízká proudová spotřeba ($150\ \mu\text{A}$), možnost připojení pomocí sběrnice I2C a 16bitová velikost výstupního slova. Převodník má 4 analogové vstupy a dle nastavení v registru lze převádět napětí buďto mezi kanálem a zemí (Single ended), nebo mezi vybranou dvojicí vstupních kanálů (Differential mode).

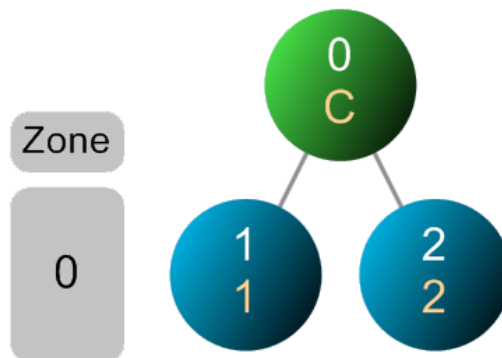
Vstupní analogový signál je impedančně oddělen od zbytku obvodu operačním zesilovačem v zapojení jako emitorový sledovač. Pro 4 kanály byly vybrány dva integrované obvody TLV2242, kde každý obsahuje dva operační zesilovače.



Obr. 19 - Výsledná podoba modulu ADC-EVAL

3.2 Návrh komunikace

Komunikace mezi jednotlivými částmi regulačního obvodu byla realizována bezdrátově s použitím IQRF transceiverů typu TR-72DA v2.14. Veškerý popis této technologie je uveden v první kapitole této práce. V úloze bylo využito síťové topologie mesh, tedy byla vytvořena IQMESH síť, která se skládá z jednoho koordinátora sítě a dvou uzlů. Komunikace probíhá v ISM pásmu 868 MHz na kanále 40 (střední frekvence = 867,15 MHz).



Obr. 20 - Schéma vytvořené mesh sítě

V koordinátorovi sítě je nahrán hardwarový profil HWP-Coordinator-STD-SPI-7xD, do obou uzlů byl nahrán hardwarový profil HWP-Node-STD-7xD, konfigurace transceiverů obsahující mimo jiné i povolení Custom DPA Handleru a zároveň pro každý uzel vlastní Custom DPA Handler. Ten v každém z uzlů definuje adresy uživatelsky vytvořených periférií (PNUM), hardwarové profily (HWPID) a implementuje rutiny jednotlivých příkazů (PCMD).

Uzel umístěný v akčním členu zpracovává pro periférii DAC-EVAL jeden příkaz, kterým se zapisuje hodnota na D/A převodník. Hodnota, která má být na D/A převodník zapsána, je uvedena ve 2 Bytech v části DATA zasílaného paketu. Jelikož je D/A převodník 12bitový, je hodnota horních 4 bitů prvního zasílaného Bytu maskována, aby nemohlo dojít k přepsání registru, kterým se určuje operační mód převodníku. Komunikace po sběrnici I2C je zahájena IQRF transceiverem a následným adresováním D/A převodníku. Poté je na D/A převodník zapsána 12bitová hodnota pro převod a komunikace je ukončena.

V následující tabulce je zaznamenán formát požadavku využívající DPA protokol pro zápis hodnoty na D/A převodník. Význam jednotlivých částí zasílaného dotazu je uveden v teoretické části v kapitole věnující se technologii IQRF, konkrétně podkapitole pojednávající o struktuře DPA protokolu (str. 15). Dotaz je uveden v hexadecimálním tvaru.

Tabulka 1: DPA požadavek pro zápis dekadické hodnoty na D/A převodník (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)	DAC val. (Hi)	DAC val. (Lo)
02	00	20	00	FF	00	01	FF

K výše uvedenému požadavku je v tabulce 1.2 zaznamenána odpověď, kterou od uzlu obdrží koordinátor. Prvních 6 Bytů zprávy je stejných, jako u zaslaného požadavku vyjma části příkazu pro periférii (PCMD), kde je k hodnotě dotazu připočtena hexadecimální konstanta 80. Dále je zaslán Byte s označením ErrN, který v případě chyby definuje příčinu, kterou lze dohledat v dokumentaci, případně uživatelsky definovanou chybu v CustomDPA Handleru. Další Byte nazvaný DPA nabývá hodnot 1 až 7F (dekadicky 1 až 128) a nese v sobě informaci o síle signálu. V následujících dvou Bytech jsou vráceny uživatelsky definovaná data, která byla uvedena v dotazu.

Tabulka 2: DPA odpověď zápisu dekadické hodnoty na D/A převodník (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)	ErrN	DPA	DAC val. (Hi)	DAC val. (Lo)
02	00	20	80	FF	00	00	5B	01	FF

Uzel ve zpětnovazebním členu rozeznává 4 příkazy. Tyto příkazy označují příslušný analogový kanál, ze kterého je převáděno napětí A/D převodníkem. PCMD 0x02 bude tedy označovat převod napětí mezi kanálem A2 a zemí. Komunikaci s převodníkem po sběrnici I2C zahajuje IQRF transceiver. Po adresování čipu ADS1115 je do jeho konfiguračního registru zapsáno nastavení převodních parametrů a tím je zároveň odstartován převod. Následně je periodicky z tohoto konfiguračního registru čteno, neboť po dokončení převodu změní převodník jeho nejvyšší bit. Po dokončení převodu je možno vyčíst z převodního registru 2 Byty obsahující digitální hodnotu aktuálního napětí na výstupu regulované soustavy.

V tabulce 1.3 je uveden formát požadavku využívající DPA protokol pro vyčtení aktuální hodnoty napětí z příslušného kanálu. Zároveň je k dotazu uvedena v tabulce 1.4 odpověď, kterou zašle uzel zpět koordinátorovi. Dotaz i odpověď jsou uvedeny v hexadecimálním tvaru.

Tabulka 3: DPA požadavek pro získání aktuální hodnoty napětí na daném kanálu (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)
01	00	20	00 až 03	0F	00

Tabulka 4: DPA odpověď aktuální hodnoty napětí na daném kanálu (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)	ErrN	DPA	ADC val. (Lo)	ADC val. (Hi)
01	00	20	80 až 83	0F	00	00	58	3A	4E

Oba uzly navíc rozeznávají příkaz PCMD 0x10, kterým je vrácena dekadická celočíselná hodnota v rozmezí 1 až 40, která je dále označena jako „level“. Dosazením získané hodnoty do uvedeného vzorce (3.1) je vypočtena aktuální hodnota napětí na výstupu LDO regulátoru, který udržuje konstantní napětí na úrovni 3 V. Tím je dosažena informace o stavu, kdy napětí Li-Pol akumulátoru začne klesat pod úroveň 3 V a je zapotřebí zvážit jeho nabití. Výstup akumulátoru, který je přiveden na vstup LDO regulátoru, může mít vyšší napětí, proto bude po dlouhou dobu získávána informace o maximální hodnotě levelu (40) a až po vybití pod napěťovou úroveň 3 V začne klesat.

$$U [V] = \frac{261,12}{127 - \text{level}} \quad (3.1)$$

V níže uvedených tabulkách jsou formáty dotazů i odpovědí pro jednotlivé uzly sloužící k získání aktuální úrovně napětí na výstupu LDO regulátoru. Požadavky pro jednotlivé uzly se liší pouze v čísle adresy daného uzlu (NADR) a hardwarovém identifikátoru daného uzlu (HWPID). Všechny uvedené zprávy jsou v hexadecimálním tvaru.

Tabulka 5: DPA požadavek pro získání aktuální hodnoty napětí na výstupu LDO regulátoru u uzlu s modulem ADC-EVAL (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)
01	00	20	10	0F	00

Tabulka 6: DPA odpověď aktuální hodnoty napětí výstupu LDO regulátoru u uzlu s modulem ADC-EVAL (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)	ErrN	DPA	Bat. level
01	00	20	90	0F	00	00	58	28

Tabulka 7: DPA požadavek pro získání aktuální hodnoty napětí na výstupu LDO regulátoru u uzlu s modulem DAC-EVAL (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)
02	00	20	10	FF	00

Tabulka 8: DPA odpověď aktuální hodnoty napětí výstupu LDO regulátoru u uzlu s modulem DAC-EVAL (hex. tvar)

NADR (Lo)	NADR (Hi)	PNUM	PCMD	HWPID (Lo)	HWPID (Hi)	ErrN	DPA	Bat. level
02	00	20	90	FF	00	00	5A	28

IQRF transceivery umístěné v uzlech indikují správnost přijatého dotazu bliknutím zelené LED. Při obdržení dotazu ve špatném tvaru (např. neznámý příkaz PCMD, neznámá periferie PNUM, špatná délka dat, špatné HWPID, atd.) dojde k bliknutí červené LED. Délka zapnutí LED je nastavena v operačním systému na 100 ms, a jelikož se obsluha vykonává na pozadí, program okamžitě pokračuje dále a nečeká na uplynutí této doby.

Koordinátor sítě také opticky indikuje zasílání dotazů. Na programátoru/debuggeru CK-USB v momentě zaslání dotazu dojde k bliknutí červené LED. Tím lze pouhým okem rozeznat, v jakém časovém okamžiku dochází k zasílání dotazů koordinátorem, zda dotaz k vybranému uzlu dorazil či nikoli, a zda měl správný tvar.

4 Identifikace soustavy

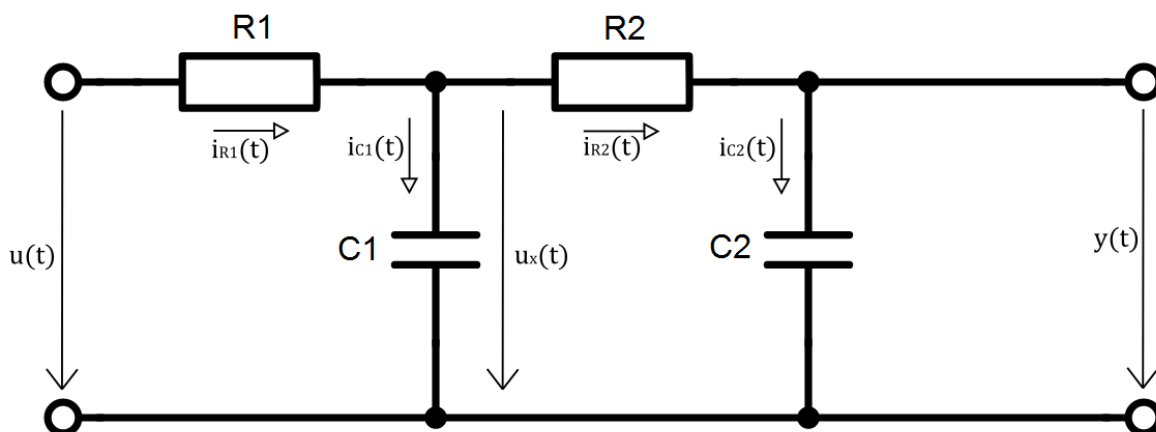
Jak již bylo zmíněno v předchozí kapitole, regulovanou soustavou pro navržené řešení je elektrický systém složený z pasivních prvků, konkrétně z rezistorů, kondenzátorů a cívek. V této úloze byla realizována soustava druhého řádu kaskádním zapojením dvou RC článků. Hodnoty pasivních součástí byly zvoleny dle normalizovaných řad následovně:

- $R1 = 4700 \, \Omega$
- $R2 = 6800 \, \Omega$
- $C1 = 470 \, \mu\text{F}$
- $C2 = 330 \, \mu\text{F}$

4.1 Analytická identifikace

Při analytické identifikaci je nutno vycházet z fyzikálních zákonů, kterými daná soustava disponuje. Tyto vnitřní procesy systému musí být známé a matematicky popsatelné. Analytická identifikace přináší výhodu zejména v možnosti vytvořit model i pro neexistující systém, tím pádem lze analyzovat dynamické vlastnosti soustavy ještě před jejím praktickým zhotovením. Obvykle však tvorba modelu vyžaduje větší časové nároky. [8]

Pro analytickou identifikaci soustavy elektrického systému je nutno aplikovat na obvod zejména Kirchhoffovy zákony, přičemž cílem je získat soustavu nezávislých rovnic. Sledovanými veličinami v elektrických obvodech jsou elektrické napětí $u(t)$ a elektrický proud $i(t)$. Dle znalosti hodnot jednotlivých veličin je následně možno rovnice upravit využitím Ohmova zákona. Pasivní prvky v obvodu jsou považovány za ideální, tedy disponují pouze jednou vlastností (odpor, kapacita, atd.) a jsou vázány pouze s jedním druhem energie. Konkrétně ideální rezistor je prvek, na kterém se elektrická energie mění v teplo, ideální kapacitor představuje prvek, který je schopen akumulovat energii elektrického pole. [8]



Obr. 21 - Schéma zapojení dvojitého RC článku

Dle prvního Kirchhoffova zákona pojednávajícím o algebraickém součtu proudů v uzlu lze odvodit následující dvojici rovnic:

$$i_{R1} = i_{R2} + i_{C1} \quad (4.1)$$

$$i_{R2} = i_{C2} \quad (4.2)$$

Po dosazení do těchto dvou rovnic s využitím Ohmova zákona je získáno dvojice rovnic:

$$\frac{u(t) - u_x(t)}{R_1} = \frac{u_x(t) - y(t)}{R_2} + C_1 \frac{du_x(t)}{dt} \quad (4.3)$$

$$\frac{u_x(t) - y(t)}{R_2} = C_2 \frac{dy(t)}{dt} \quad (4.4)$$

Vyjádřením napětí na kondenzátoru C1 z druhé rovnice (4.4) a následným dosazením do první rovnice (4.3) je dosaženo vztahu ve tvaru:

$$\frac{u(t)}{R_1} - \frac{y(t)}{R_1} - \frac{R_2 C_2}{R_1} \cdot \frac{dy(t)}{dt} = C_2 \frac{dy(t)}{dt} + R_2 C_1 C_2 \frac{d^2 y(t)}{dt^2} + C_1 \frac{dy(t)}{dt} \quad (4.5)$$

Po roznásobení a drobných úpravách vzejde finální podoba diferenciální rovnice:

$$u(t) = R_1 R_2 C_1 C_2 \cdot y''(t) + (R_1 C_1 + R_2 C_2 + R_1 C_2) \cdot y'(t) + y(t) \quad (4.6)$$

Přenos soustavy v Laplaceově transformaci je poté dán poměrem obrazů výstupního a vstupního signálu:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{R_1 R_2 C_1 C_2 \cdot s^2 + (R_1 C_1 + R_2 C_2 + R_1 C_2) \cdot s + 1} \quad (4.7)$$

Po dosazení hodnot pasivních součástí je dosažen přenos ve tvaru:

$$G(s) = \frac{1}{4,957 \cdot s^2 + 6,004 \cdot s + 1} \quad (4.8)$$

Z tvaru výsledného přenosu lze vyčíst zesílení a časové konstanty:

- $K = 1$
- $T_{p1} = 5,016$
- $T_{p2} = 0,988$

4.2 Experimentální identifikace

Metody experimentální identifikace vyžadují existenci zkoumaného systému a možnost provádět s ním experimenty. U tohoto druhu identifikace je potřebná znalost fyzikální struktury nižší, než u metod analytické identifikace, neboť vnitřní procesy systému nemusí být známy. Při identifikaci je pohlíženo na soustavu jako na černou skříňku a po přivedení deterministického signálu na vstup soustavy je zaznamenávána odezva na výstupu soustavy. Tato odezva buď přímo představuje neparametrickou charakteristiku soustavy, nebo je možno pomocí vhodných metod neparametrickou charakteristiku sestavit. Z této charakteristiky lze následně matematickými metodami získat přenos soustavy. [8]

Na regulovanou soustavu byl přiveden skokový signál o velikosti napětí 3 V. Za pomoci algoritmu naprogramovaného v řídicím systému REX bylo napětí na výstupu soustavy vzorkováno s periodou 500 ms. Tato naměřená data byla dále importována do prostředí Matlab, kde byl pomocí

System Identification Toolboxu zjištěn přenos identifikované soustavy. Vstupními parametry pro identifikaci byly vektory vstupního a výstupního signálu, perioda vzorkování a předpokládaný řád soustavy.

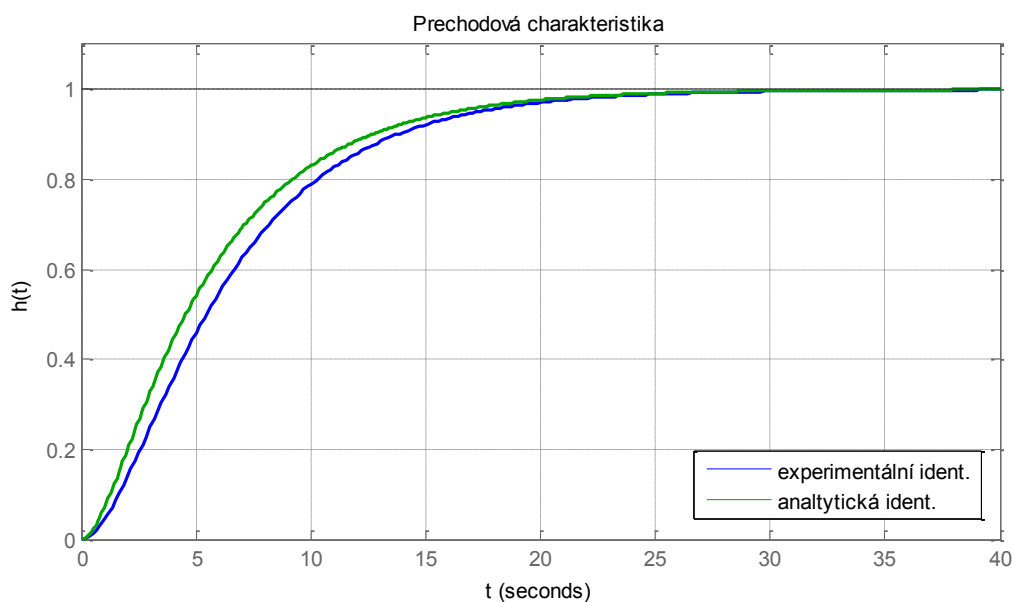
Mezi průběhem změřeným a identifikovaným není téměř žádný rozdíl. Soustava byla v Matlabu identifikována dle best fit přesnosti na 99,13 %. Odhad parametrů zesílení a časových konstant dopadl takto:

- $K = 1$
- $Tp1 = 5,004$
- $Tp2 = 1,815$

Výsledný přenos soustavy je tedy:

$$G(s) = \frac{K}{(Tp1 \cdot s + 1)(Tp2 \cdot s + 1)} = \frac{1}{9,084 \cdot s^2 + 6,82 \cdot s + 1} \quad (4.9)$$

V porovnání analyticky a experimentálně identifikované soustavy lze pozorovat, že větší časová konstanta systému je téměř stejná, zatímco nižší časová konstanta se výrazně liší. Důvod tohoto nesouladu může být v reálných hodnotách pasivních prvků, kdy v analytickém odvození se uvažovalo s normalizovanou hodnotou, která může být ovlivněna přesností daného prvku. Zároveň může být příčina v tom, že u analytické identifikace je uvažována nekonečně velká hodnota impedance na vstupu i výstupu soustavy, avšak v reálném provedení tomu tak není.



Obr. 22 - Přechodová charakteristika analyticky a experimentálně identifikované soustavy

K dalšímu postupu v práci, především pak k syntéze regulátoru, byl vybrán přenos určený experimentální identifikací.

5 Syntéza regulátoru

Pod pojmem syntéza regulátoru je myšleno navržení vhodného regulátoru pro zadanou soustavu a stanovení jeho parametrů. Účelem regulátoru je dosažení a udržení nulové regulační odchylky, případně udržení v předepsaných mezích. Pro správnou syntézu regulátoru je nutno znát dynamické vlastnosti soustavy. Tyto vlastnosti byly získány na základě analytické a experimentální identifikace prováděné v předchozí kapitole.

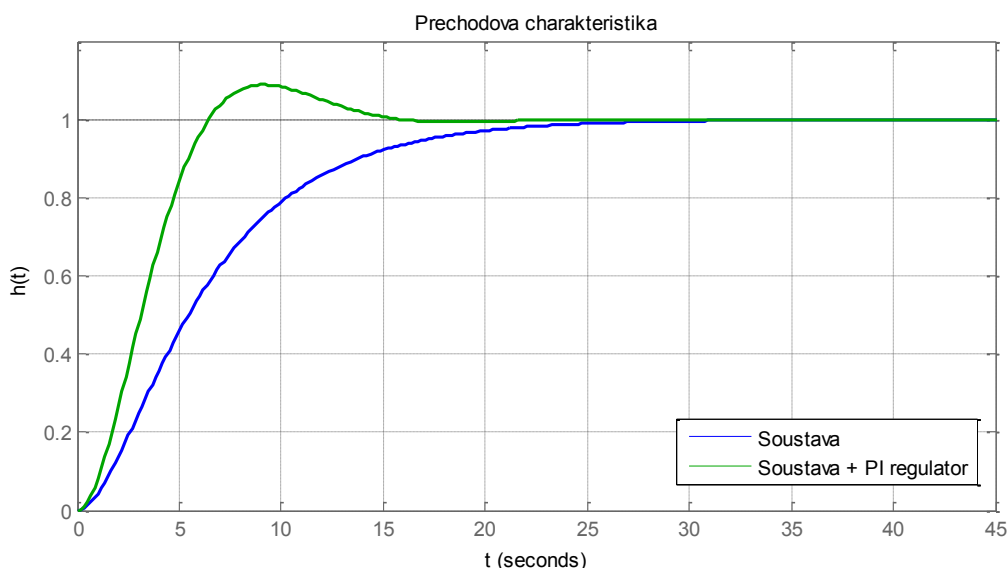
Pro regulaci soustavy byl zvolen regulátor typu PI. Konstanty regulátoru byly navrženy s pomocí softwarového nástroje Matlab, kde je k dispozici funkce *pidtune*. Zadáním přenosu identifikované soustavy z předchozí kapitoly (vzorec 4.9) a definováním požadovaného typu regulátoru do funkce jsou vypočteny a vráceny konstanty přenosu PI regulátoru v paralelním tvaru. Vzorec tohoto přenosu po dosazení vypočtených konstant je následující:

$$G_r(s) = K_p + K_i \cdot \frac{1}{s} = 1,76 + 0,361 \cdot \frac{1}{s} \quad (5.1)$$

Po zařazení regulátoru do regulačního obvodu lze přenos uzavřené smyčky obvodu odvodit podle následujícího vzorce:

$$G_w(s) = \frac{G_r(s) \cdot G(s)}{1 + G_r(s) \cdot G(s)} = \frac{1,755s + 0,361}{9,084s^3 + 6,82s^2 + 2,755s + 0,361} \quad (5.2)$$

Při srovnání odezvy na jednotkový skok samotné soustavy a soustavy zařazené do regulačního obvodu v prostředí Matlab lze vidět, jak navržený regulátor ovlivňuje podobu výstupu regulované veličiny. Parametry regulátoru lze dále ladit a vykreslovat průběhy přechodových charakteristik k dosažení požadovaného výsledku, případně lze k návrhu regulátoru využít matematických metod pro to určených. Tato práce se však primárně nezabývá procesem syntézy regulátoru, a proto pro další postup postačí výsledek, který byl získán pomocí funkce *pidtune* v prostředí Matlab.



Obr. 23 - Srovnání přechodové charakteristiky soustavy bez regulátoru a s regulátorem

Následující tabulka shrnuje vlastnosti přechodové charakteristiky regulované soustavy a přechodové charakteristiky po zařazení do regulačního obvodu.

Tabulka 9: Srovnání vlastností odezev na jednotkový skok

	Bez regulátoru	S regulátorem
Doba náběhu	12,1245 s	4,3205 s
Doba ustálení	21,8339 s	13,7913 s
Překmit	0 %	8,8523 %

Ve vývojovém prostředí RexDraw je k dispozici 6 typů PID regulátoru. Ty nabízí i pokročilé funkce jako například zabudovanou funkci automatického nastavení parametrů na základě odezvy na obdélníkový impuls (PIDMA), možnost automatického nastavení parametrů na základě změření jednoho bodu frekvenční charakteristiky s vhodným fázovým zpožděním (PIDAT) atd. Stojí za to zmínit, že uvedené příklady regulátorů vyžadují vyšší licenci produktu REX, než je standardní verze.

Při návrhu regulátoru pro tuto práci bylo využito základního bloku s názvem PIDU. Konstanty jednotlivých složek se zde zadávají v ideálním tvaru, proto je nutné navržené konstanty v paralelním tvaru přepočítat. Výsledný přenos vypadá následovně:

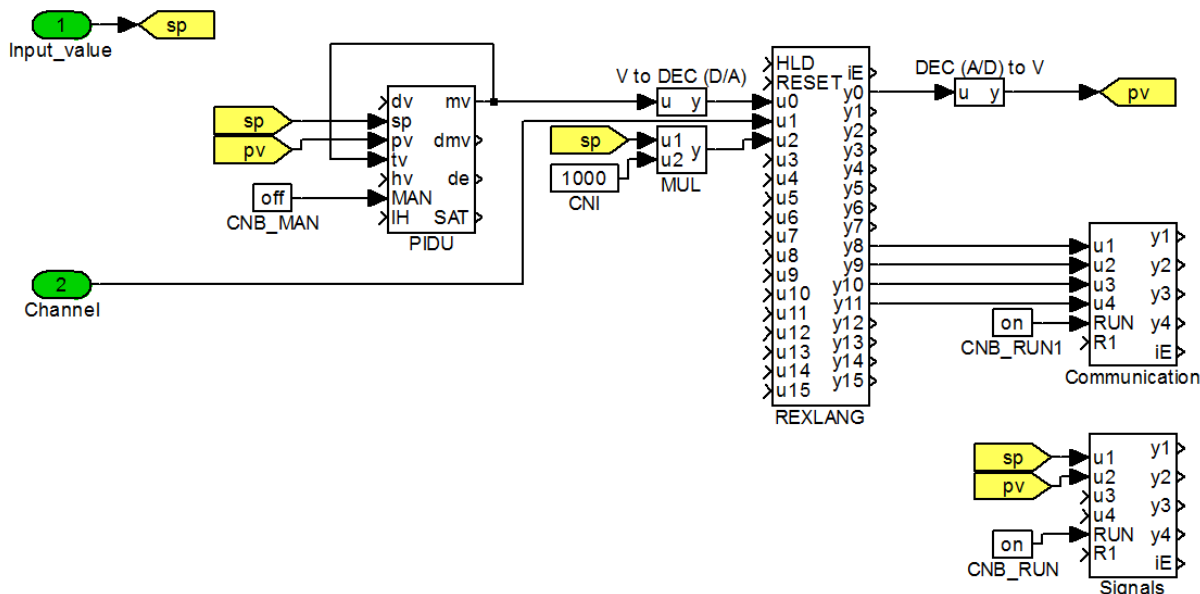
$$G_r(s) = K_p \cdot \left(1 + \frac{1}{T_i s}\right) = 1,76 \cdot \left(1 + \frac{1}{4,8753 \cdot s}\right) \quad (5.3)$$

Dále je v tomto typu regulátoru možnost zadat váhové faktory pro proporcionální a derivační složku. V práci byly váhy ponechány ve tvaru, kdy nemají na zadané složky vliv. Důležité je nastavení časové konstanty vysledování (označenou v PIDU jako tt). Správné nastavení konstanty zaručí bezrázový přechod při přepínání regulátoru mezi manuálním a automatickým režimem a zároveň jeho správnou funkci při saturaci výstupu. Nutno podotknout, že tento parametr má význam pouze u regulátorů s integrační složkou, což je případ této práce. Doporučené nastavení hodnoty pro PI regulátor je dvojnásobek integrační konstanty T_i . Toto doporučení bylo dodrženo a hodnota parametru časové konstanty vysledování tt byla nastavena na 9,75.

Dalším parametrem v nastavení regulátoru je dolní a horní limit akčního zásahu. Jelikož D/A převodník na modulu DAC-EVAL má rozsah výstupního napětí 0 až 3,3 V, není zapotřebí, aby na výstupu regulátoru byly vyšší nebo nižší hodnoty akčního zásahu, než je tento rozsah. Z tohoto důvodu byly tyto úrovně napětí zapsány do parametrů $hilim$ a $lolim$ regulátoru PIDU.

Funkční blok PIDU disponuje 7 vstupy a 4 výstupy, jak lze vidět na obrázku 24. Prvním vstupem je signál dopředné vazby (dv). Tato vazba v realizovaném systému není, proto je vstup ponechán nezapojený. Druhým vstupem je požadovaná hodnota (sp). Tato hodnota je navázána na vizualizaci, kde je uživatelsky zadávána pomocí posuvníku. Jako řízená veličina (pv) je přivedena aktuální hodnota regulované veličiny, která je získávána ze zpětnovazebního členu. Dalším vstupem je veličina pro vysledování (tv), kde je přivedena zpětnou vazbou hodnota akčního zásahu z výstupu PIDU bloku. Regulátor je přivedením konstanty ' off ' na vstup MAN nastaven do automatického režimu, tím pádem vstup definující hodnotu výstupu v manuálním režimu (hv) je ponechán nezapojen. Jediným použitým výstupem funkčního bloku PIDU je aktuální hodnota akčního zásahu (mv), která je následně pomocí bezdrátové IQRF komunikace přenášena do akčního členu a přiváděna na vstup regulované soustavy.

Celé schéma regulačního algoritmu vytvořené v programu RexDraw je zobrazeno na následujícím obrázku. Jedná se o soubor s příponou *.mdl* v prostředí REX označovaný jako *task*. Ten je připojen k výstupu *Level0* bloku Real-Time Executive, kde je nastavena perioda regulačního algoritmu 250 ms.



Obr. 24 - Schéma regulačního algoritmu v programu RexDraw

V schématu jsou zelenou barvou označeny prvky s názvem Input Port. Jedná se o vstup požadované hodnoty a kanálu, které jsou zadávány uživatelem přes vizualizační rozhraní. Žlutou barvou jsou označeny klíčové signály regulačního obvodu. Konkrétně jde o požadovanou hodnotu (*sp*) a regulovanou veličinu (*pv*).

Pro převod reálné hodnoty akčního zásahu na hodnotu v rozsahu D/A převodníku je použit blok lineární interpolace. Stejný blok je také použit pro převod dekadické hodnoty v rozsahu A/D převodníku na reálnou hodnotu regulované veličiny. Parametry zadané v bloku jsou rozsahy vstupní a výstupní veličiny. Například pro převod akčního zásahu je vstupní rozsah 0 až 3,3 V, výstupní mez odpovídá 12bitovému rozsahu D/A převodníku, tedy rozmezí 0 až 4095.

Signály požadované hodnoty a regulované veličiny jsou přivedeny do prvku Real-time trend s pojmenováním Signals. Do dalšího trendu s názvem Communication jsou přivedeny signály vystupující z bloku REXLANG. Jedná se o počet přijatých a odeslaných Bytů a počet ztracených odpovědí od jednotlivých uzlů. Oba trendy jsou použity pro vizualizační účely a jejich grafická podoba je popsána v kapitole Vizualizace.

5.1 Dotazovací algoritmus

Systém dotazování jednotlivých uzlů je řízen regulačním algoritmem běžícím v runtime jádru RexCore v cílovém zařízení Raspberry Pi 3. Regulační algoritmus běží s periodou cyklu 250 ms. K naprogramování komunikace byl použit volně programovatelný blok REXLANG, který implementuje algoritmus definovaný skriptovacím jazykem. Syntaxe skriptovacího jazyka vychází z jazyka C, avšak nepodporuje například řadu datových typů, operátorů či příkazů.

Výhodou tohoto programovatelného bloku je, že pomocí něj lze přistupovat k sériové lince. Tím lze komunikovat s IQRF koordinátorem, který poté komunikuje s celou sítí. Důležitým krokem je správné nastavení portu, na kterém se bude komunikovat. Programátor/debugger CK-USB s IQRF transceiverem je k Raspberry Pi 3 připojen jako `ttyACM0`. Do parametrů bloku REXLANG je tedy uvedena cesta `/dev/ttyACM0`.

Blok REXLANG disponuje vstupně-výstupním rozhraním, pomocí kterého ho lze propojit s jinými funkčními bloky a tím ho zařadit do struktury. Jak počet vstupů, tak počet výstupů je omezen maximálně na 16 signálů. V níže uvedeném seznamu je přehled všech vstupů a výstupů, které byly pro realizaci dotazovacího algoritmu využity.

Vstupy:

- *dacValue* – Aktuální hodnota akčního zásahu v rozsahu D/A převodníku
- *channel* – Analogový kanál A/D převodníku, ze kterého je snímáno napětí
- *presetValue* - Požadovaná hodnota

Výstupy:

- *adcValue* – Aktuální hodnota regulované veličiny v rozsahu A/D převodníku
- *adcSignal* – Síla signálu obdrženého od zpětnovazebního uzlu
- *dacSignal* – Síla signálu obdrženého od akčního uzlu
- *adcBattery* – Aktuální stav baterie zpětnovazebního uzlu
- *dacBattery* – Aktuální stav baterie akčního uzlu
- *receivedBytes* – Počet Bytů přečtených z bufferu sériového portu v daném cyklu
- *sentBytes* – Počet Bytů zaslaných po sériové lince v daném cyklu
- *noResponseADInRow* – Počet dotazů bez odpovědi v řadě od zpětnovazebního uzlu
- *noResponseDAInRow* – Počet dotazů bez odpovědi v řadě od akčního uzlu
- *noResponseAD* – Celkový počet dotazů bez odpovědi od zpětnovazebního uzlu
- *noResponseDA* – Celkový počet dotazů bez odpovědi od akčního uzlu
- *responseAD* – Celkový počet obdržených odpovědí od zpětnovazebního uzlu
- *responseDA* – Celkový počet obdržených odpovědí od akčního uzlu

Princip funkce událostmi řízeného regulátoru nespočívá v nastavení PI regulátoru, nýbrž v principu dotazování. Je-li hodnota regulační odchylky nulová, případně leží-li v předepsaném pásmu, není zapotřebí generovaný akční zásah neustále zasílat do akčního členu. V tomto případě je pouze potřeba kontrolovat hodnotu regulované veličiny, zda odpovídá požadované hodnotě. Tím pádem je zapotřebí dotazovat zpětnovazební člen s požadavkem o zaslání aktuální hodnoty regulované veličiny.

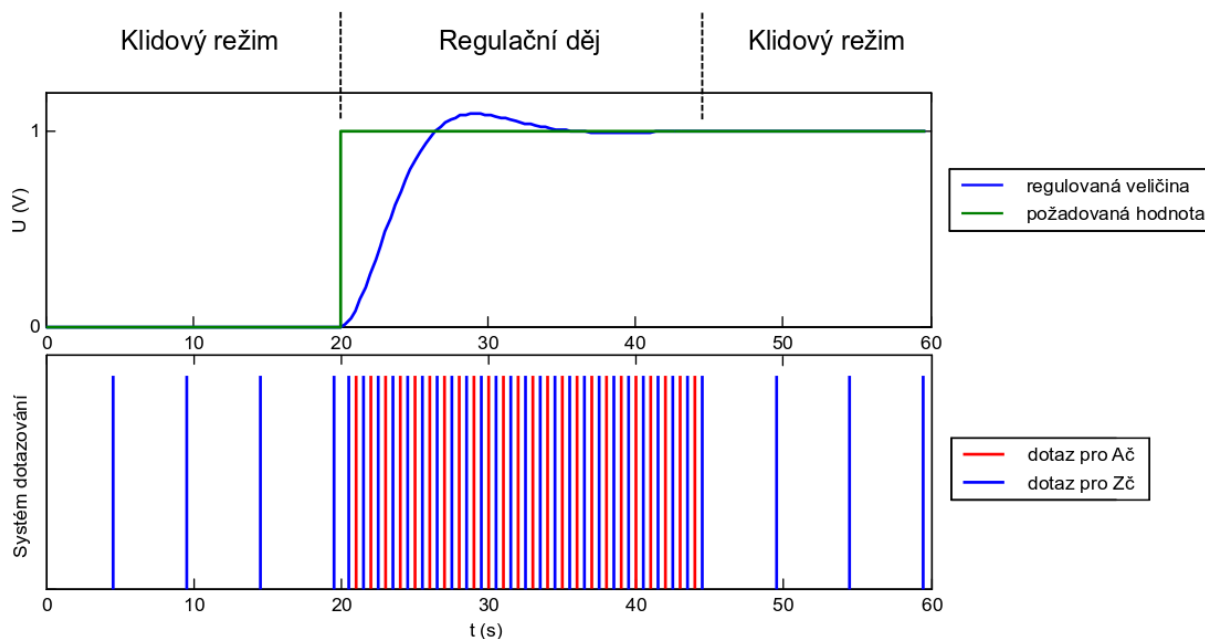
Jakmile však hodnota regulační odchylky přesáhne meze předepsaného pásma, je zapotřebí provést regulační proces. Z uvedeného popisu vyplývá, že v řídicím algoritmu mohou nastat tyto 2 stavy:

- **Regulační děj** – hodnota regulované veličiny neleží v předepsaném pásmu, a tudíž je vyžadována regulace.
- **Klidový režim** – hodnota regulované veličiny leží v tolerančním pásmu požadované hodnoty.

Jak již bylo zmíněno, regulační děj nastává v případě, kdy regulační odchylka překročí meze předepsaného pásma. Situace, kdy k překročení daného pásma dojde, jsou tyto:

- Uživatelským zásahem se změní požadovaná hodnota
- Regulovaná veličina opustí předepsané pásmo požadované hodnoty

V prvním uvedeném příkladu dojde ke změně regulační odchylky tím, že se cílevědomě změní požadovaná hodnota. V druhém případě dojde vlivem poruchy (okolního prostředí, fyzického zásahu, poklesu napětí atd.) k ovlivnění regulované veličiny, čímž dojde k opuštění předepsaného pásma. V obou těchto případech je nutno přejít z klidového režimu do regulačního děje.



Obr. 25 - Znázornění stavů v řídicím algoritmu a systém dotazování

Nastanou-li podmínky spouštějící regulační děj, sekvence vykonávání regulačního algoritmu vypadá následovně:

- **0. cyklus** ($t = 0$ ms) – Dotaz na hodnotu regulované veličiny ze zpětnovazebního uzlu
- **1. cyklus** ($t = 250$ ms) – Dekódování odpovědi, zápis hodnoty akčního zásahu na akční člen
- **2. cyklus** ($t = 500$ ms) – Dekódování odpovědi, dotaz na hodnotu regulované veličiny ze zpětnovazebního uzlu
- **3. cyklus** ($t = 750$ ms) – Stejný jako 1. cyklus

Probíhá-li regulační děj, koordinátor zasílá požadavky střídavě na jednotlivé uzly. Nejprve je vyžádána aktuální hodnota regulované veličiny ze zpětnovazebního uzlu (0. cyklus), na základě regulační odchylky je vypočten akční zásah a ten je následně zaslán do akčního členu (1. cyklus). V dalším cyklu (2. cyklus) je z odpovědi akčního členu zpracována informace, zda zpráva obsahující akční zásah dorazila správně a opět je zaslán požadavek pro aktuální hodnotu regulované veličiny.

Aby bylo možno ukončit regulační děj, nestačí, aby se hodnota regulované veličiny ocitla v předepsaném pásmu. To z toho důvodu, že při kmitavém charakteru přechodové charakteristiky se regulovaná veličina ocitne v předepsaném pásmu vícekrát a nemusí to značit ustálený stav. Proto je třeba zabezpečit, aby se regulační děj ukončil tehdy, kdy již regulovaná veličina pásmo neopustí. V řídicím algoritmu, konkrétně v bloku REXLANG, je tato situace ošetřena pomocí posuvného registru. Ten je vytvořen pro 20 hodnot a porovnává své minimum a maximum. Jestliže absolutní rozdíl těchto hodnot nepřesahuje úroveň 0,01 V, je regulační děj ukončen.

V klidovém režimu je co 20 cyklů, tedy s periodou 5 s, dotazována hodnota regulované veličiny ze zpětnovazebního uzlu. Zároveň je s periodou 120 s (každý 480. cyklus) dotazována aktuální hodnota napětí na výstupu LDO regulátoru, přičemž perioda dotazů mezi zpětnovazebním a akčním uzlem je 60 s.

Zavedením uvedeného principu dotazování je zajištěna mnohem menší proudová spotřeba IQRF transceiverů, neboť není nutno přijímat velké množství zpráv, a zároveň nižší spotřeba periferních modulů na jednotlivých uzlech. Tím se razantně zvyšuje výdrž Li-Pol akumulátorů na daných uzlech. Zároveň je v klidovém režimu při neobdržení odpovědi od jednoho z uzlů prostor pro opakování dotazu. Tato funkce však v práci nebyla implementována.

6 Vizualizace

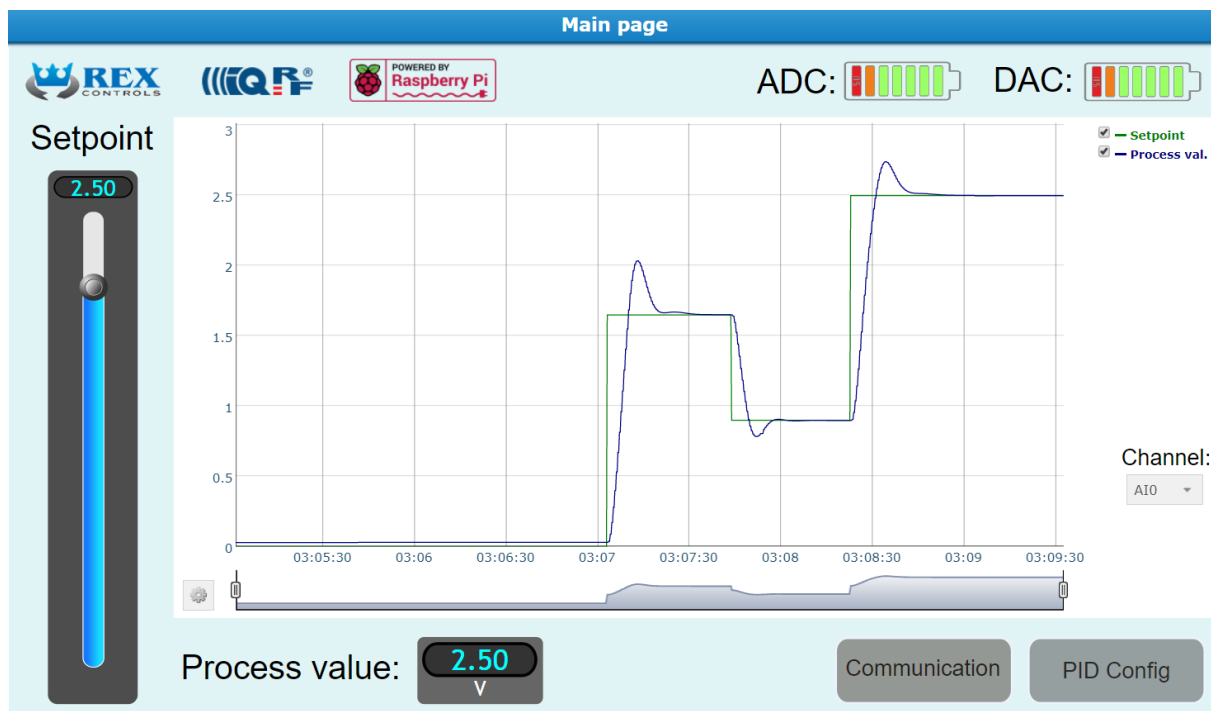
Vizualizační rozhraní bylo vytvořeno v programu RexHMI Desinger, který vychází z vektorového grafického editoru Inkscape. Při kompilaci celého projektu v programu RexDraw jsou na základě vytvořených vizualizačních obrazovek vygenerovány kaskádní styly CSS a soubory v jazyce HTML, jedná-li se o statickou vizualizaci, a v jazyce JavaScript, jedná-li se o dynamické prvky jako jsou grafy, indikátory atd. Vytvořená vizualizace se dohromady skládá ze 4 obrazovek:

- Úvodní obrazovka
- Zobrazení signálů (Main page)
- Záznam komunikace (IQRf Communication)
- Nastavení PID regulátoru (PID Configuration)

Úvodní obrazovka se zobrazí pouze při prvním načtení vizualizace v internetovém prohlížeči a nese v sobě informace o názvu práce a autorovi. Kliknutím na jakékoliv místo na obrazovce dojde k přepnutí na obrazovku zobrazující vstupní a výstupní signály.

Zde se nachází ovládací prvek s popisem *Channel* sloužící k výběru analogového kanálu pro převod napětí modulem ADC-EVAL. Prvek je provázán s blokem Input Port v řídicí aplikaci vytvořené v RexDraw. Dále se zde nachází horizontální posuvník pod názvem *Setpoint* pro zadávání požadované hodnoty, kterou lze zadávat v rozmezí 0 až 3 V. Prvek je opět provázán s blokem Input Port řídicí aplikace.

Pro indikaci aktuální hodnoty regulované veličiny je použit displej. Zároveň je vývoj regulované veličiny a požadované hodnoty v čase zaznamenán v trendu. Paměť bufferu trendu je nastavena na 1000 vzorků.



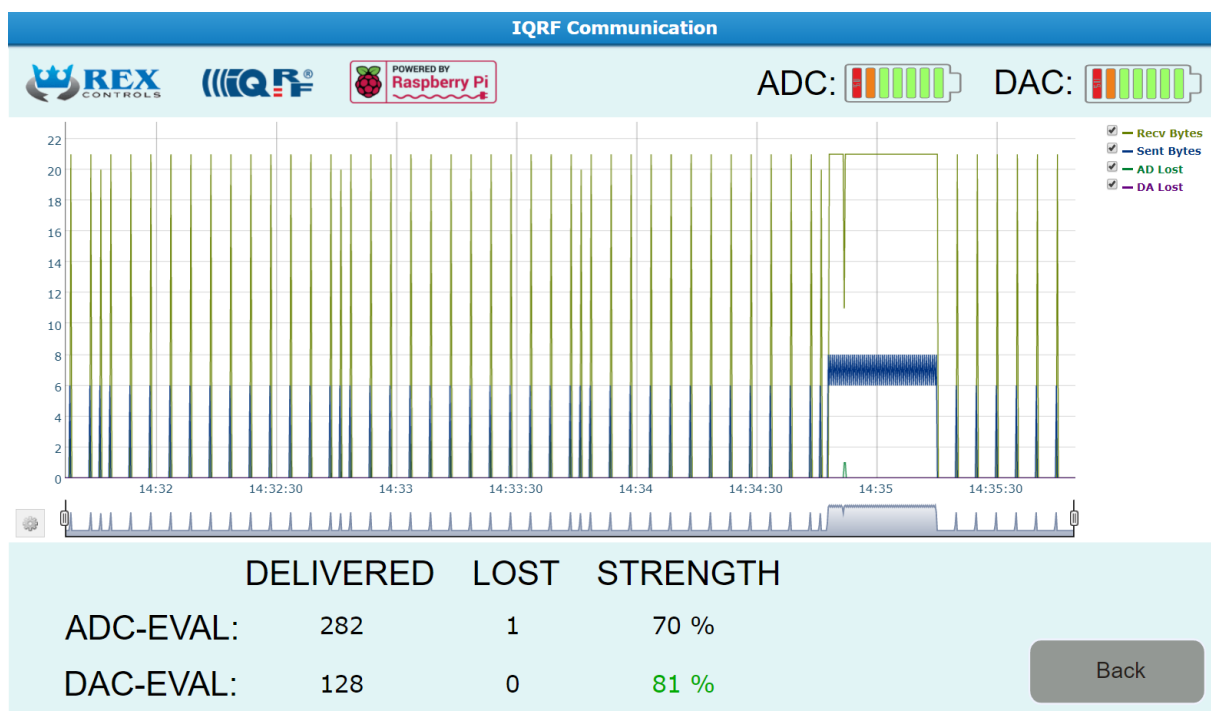
Obr. 26 - Vizualizace regulačního obvodu

Na obrazovce záznamu komunikace lze v trendu sledovat počet zasílaných a přijímaných Bytů koordinátorem během časového úseku. Je-li zasláno 8 B dat, jedná se o dotaz pro akční člen, v případě zaslání 6 B dat jde buď o dotaz pro zpětnovazební uzel na aktuální hodnotu regulované veličiny, nebo o dotaz na stav baterie. Obdržená data o velikosti 21 B nebo 20 B znamenají úspěšnou odpověď od jednoho z uzlů, při obdržení dat o velikosti 11 B došlo ke ztrátě odpovědi od uzlu a tato data znamenají pouze potvrzení o správnosti předchozího dotazu.

Zároveň je v tomto vizualizačním okně číselná informace o počtu přijatých odpovědí od jednotlivých uzlů v síti, počtu ztracených odpovědí a informace o síle signálu. Síla signálu je vyčítána z každé odpovědi od příslušného uzlu, kde se tato informace zasílá v 1 Bytu a nabývá hodnot 0x01 až 0x7F. Tato hodnota je poté přepočtena na škálu 1–100 %.

Mimo počet zasílaných a přijatých Bytů lze v trendu sledovat též záznam ztracených odpovědí od jednotlivých uzlů. Dojde-li ke ztrátě více odpovědí v řadě, tato hodnota roste. Jakmile je od daného uzlu obdržena odpověď, hodnota je vynulována. Na základě této informace je možno detekovat poruchu daného uzlu.

Veškeré vizualizační proměnné v tomto vizualizačním okně jsou navázány na signály řídicího algoritmu vytvořeného v programu RexDraw. Jedná se zejména o výstupy z volně programovatelného bloku REXLANG. Seznam těchto výstupů je zároveň s popisy uveden v předchozí kapitole.



Obr. 27 - Vizualizace záznamu komunikace

Poslední obrazovka obsahuje ovládací prvky pro zadávání parametrů k nastavení PID regulátoru. V základním nastavení jsou parametry vyplněny tak, jak byly navrženy v přechozí kapitole. Při běhu celého systému lze měnit jednotlivé konstanty a pozorovat trend regulované veličiny. Každý numerický ovládací prvek na této obrazovce je propojen s příslušným parametrem funkčního bloku PIDU, který byl použit v regulačním algoritmu.

PID Configuration

ADC: [Battery Icon] DAC: [Battery Icon]

Type of regulator:

K:

Ti:

Td:

High limit:

Low limit:

Derivating filtering const:

Weighting - propotional:

Weighting - derivative:

Tracking time const:

Obr. 28 - Vizualizace nastavení regulátoru

V pravém horním rohu je na všech obrazovkách indikátor stavu baterie u jednotlivých uzlů (viz. Obr. 26, 27, 28). Indikátor zobrazující stav baterie je rozdělen do 7 bloků, kde plný stav odpovídá napětí 3 V a prázdný stav úrovni 2,07 V. Pohybuje-li se napětí mezi těmito dvěma hranicemi, je poměrově zobrazen daný počet bloků.

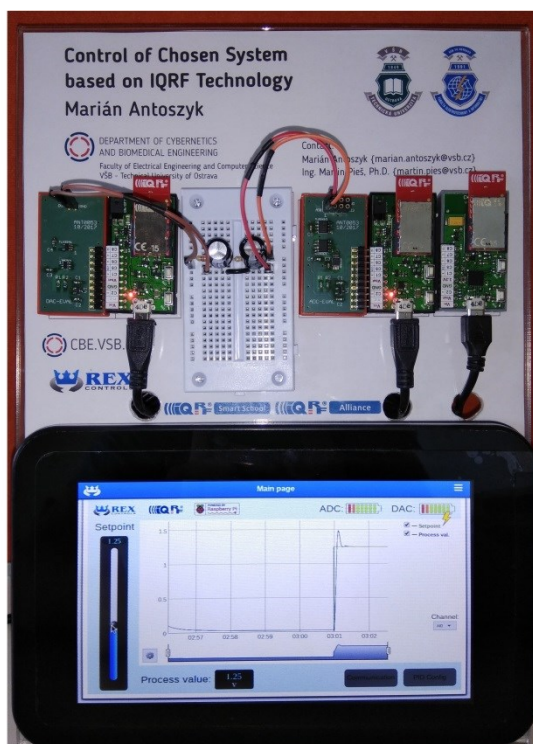
Celá vizualizace byla nahrána do cílové platformy Raspberry Pi 3. Vizualizace je spustitelná přes webový prohlížeč. Doporučenými prohlížeči jsou Google Chrome a Mozilla Firefox, neboť v jiných prohlížečích může dojít ke špatnému zobrazení některých prvků, případně nebudou prvky zobrazeny vůbec. Vizualizace běží na cílovém zařízení, konkrétně na portu 8008. IP adresa zařízení byla nastavena staticky na 192.168.10.10. Dále bylo cílové zařízení Raspberry Pi 3 nastaveno jako Wi-Fi Access Point, tudíž k vizualizaci lze přistupovat z notebooku, mobilního telefonu, tabletu či jiného zařízení umožňujícího Wi-Fi připojení. Přístupové údaje byly zvoleny takto:

- SSID: *IQRF_REX*
- Heslo: *REX_IQRF_2018*

Pro přístup k vizualizaci je nutno zadat do webového prohlížeče adresu ve tvaru <http://192.168.10.10:8008/>. Po načtení adresy naběhne logovací systém REX, kde stačí zadat pouze uživatelské jméno *admin* a připojit se. Následně je zobrazeno úvodní vizualizační okno.

7 Zhodnocení výsledků

Celý realizovaný systém byl pro prezentační účely přimontován na plastový stojan, čímž je umožněno ho jednoduše přenášet z místa na místo. K systému je vhodné používat také externí klávesnici pro zadávání číselných vstupů k nastavení regulátoru.



Obr. 29 - Vytvořený prezentační stojan

Časová prodleva mezi zasláním dotazu a obdržení odpovědi je klíčovým parametrem pro možnost použití bezdrátové technologie v regulačním obvodu. V realizovaném řešení se tato hodnota pohybovala mezi 150 až 210 ms. Perioda řídicího algoritmu byla na základě těchto hodnot nastavena na 250 ms, přičemž úspěšnost komunikace byla vyšší než 98 %. V návaznosti na zmíněnou časovou prodlevu a periodu řídicího algoritmu je nutno brát v potaz také časovou konstantu regulované soustavy. V této práci vyšla časová konstanta zjištěná pomocí experimentální identifikace 9,084 s. Jelikož periodu regulačního algoritmu už nelze příliš snižovat, není možno regulovat soustavy s malou časovou konstantou. Velikost vzorkovací periody regulované veličiny má totiž podstatný vliv na stabilitu regulačního obvodu i na jeho další vlastnosti.

Alternativa bezdrátového řešení může být nevýhodná u regulačních obvodů, kde často dochází k překročení předepsaných mezí pro akční zásah. Tato situace může nastávat častou změnou požadované hodnoty regulované veličiny, případně vlivem poruch působících na regulovanou veličinu. Tím pádem dochází k častému spouštění regulačního děje, který má za následek větší proudovou spotřebu na jednotlivých uzlech. Pokud jsou tyto uzly napájeny pouze z Li-Pol akumulátoru, dochází k jejich rychlejšímu vybíjení a je nutné je častěji dobíjet. V případě uzlů napájených z externího zdroje napětí však tato nevýhoda neplatí.

Práce se mimo jiné zabývala identifikací a návrhem regulátoru pro jednu konkrétní regulovanou soustavu. Jelikož je praktické řešení práce navrženo pro soustavy tvořené elektrickými pasivními součástkami, je možno sestavit libovolnou soustavu, opět provést kroky vedoucí k identifikaci a syntéze regulátoru a následně zhodnotit možnost regulace zvolené soustavy pomocí bezdrátové technologie IQRF. Celý postup lze udělat bez zásahu do softwarové části řešení, neboť typ a parametry regulátoru lze měnit z vizualizace. Zásah by však musel být proveden v případě, že by bylo zapotřebí měnit periodu cyklu řídicího systému REX, která byla v práci stanovena na 250 ms.

Softwarovým rozšířením stávajícího řešení by mohla být implementace zasílání opakovaných dotazů v případě neobdržení odpovědi do definovaného časového intervalu. Znamenalo by to úpravu dotazovacího algoritmu ve volně programovatelném bloku REXLANG. Tato funkcionality by byla jistě přínosem v klidovém režimu, nicméně také při regulačním ději by bylo vhodné uvážit její výhody a nevýhody.

Dalším realizovatelným rozšířením může být ošetření poruchových stavů, kdy od jednoho z uzlů přestanou kontinuálně chodit odpovědi. Informace o neobdržené odpovědi jsou již v řešení implementovány, nicméně jsou pouze zaznamenávány do trendu a číselných indikátorů, systém na ně nijak nereaguje. Vyhodnocení těchto informací by mohlo být formou alarmu zobrazeného ve vizualizaci systému.

Hardwarovými možnostmi vylepšení stávajícího řešení se jeví například odstranění programátoru/debuggeru CK-USB-04, ve kterém je umístěn IQRF transceiver sloužící jako koordinátor sítě. Místo toho může být použit adaptér KON-RASP-01, do kterého lze zasunout IQRF transceiver a tento adaptér připojit přes GPIO piny přímo k platformě Raspberry Pi. Tím by komunikace mezi řídicí jednotkou a koordinátorem sítě probíhala po sběrnici SPI.

Jako další možnost vývoje lze uvést zejména nasazení realizovaného řešení na jinou fyzikální soustavu. Jako realizovatelná možnost se jeví například regulace teploty či regulace výšky hladiny. Stále však musí být u daných soustav brána v potaz jejich časová konstanta v porovnání s časovou prodlevou mezi zasláním dotazu a obdržáním odpovědi při bezdrátové komunikaci.

V případě použití více uzlů v IQMESH síti, zejména pak zpětnovazebních uzlů (senzorická část), by bylo s největší pravděpodobností zapotřebí přepřetrat princip bezdrátové komunikace. Dotazování každého uzlu samostatně se v tomto případě jeví jako neefektivní. IQRF však umožňuje hromadné vyčítání dat z uzlů prostřednictvím FRC (Fast Response Command). Tím je možno zasláním jednoho požadavku získat odpovědi od všech dotázaných uzlů najednou. Implementace tohoto principu by vyžadovala zásah do CustomDPA Handlerů jednotlivých uzlů a zároveň přepřetrat dotazovacího algoritmu v řídicím systému REX.

Závěr

Cílem bakalářské práce bylo využití bezdrátové technologie IQRF jako alternativy pro přenos signálů při regulaci soustavy. Jako soustava byl zvolen elektrický systém skládající se z dvojitého RC článku.

V rámci řešení bakalářské práce byly vytvořeny periferní moduly, které jsou kaskádně připojeny k IQRF node kitům. Jeden modul v regulačním obvodu působí jako akční člen, druhý modul se chová jako snímač regulované veličiny.

Komunikace mezi jednotlivými částmi regulačního obvodu byla realizována sestavením síťové topologie mesh (IQMESH). Využitím DPA protokolu používaného v IQRF systémech byly definovány formáty dotazů a odpovědí.

V práci se podařilo vytvořit regulační obvod, ve kterém není na výstupu poznat, že jsou hodnoty regulované veličiny a akčního zásahu předávány pomocí bezdrátové komunikace místo běžných vodičů. Zároveň se povedlo úspěšně implementovat událostmi řízený regulátor, který se pro bezdrátové řešení hodí.

Vizualizace celého systému byla vytvořena pomocí RexHMI Designeru. Vygenerované soubory CSS, HTML a JS byly nahrány do cílového zařízení, a tedy celá vizualizace běží na localhostu na portu 8008. Cílová platforma Raspberry Pi 3 byla nastavena jako Wi-Fi Access Point, tudíž se lze k vizualizaci připojit přes různá zařízení.

Realizovaný systém byl umístěn na prezentační panel a umožňuje praktickou ukázkou technologií IQRF a REX. Dále bude panel sloužit jako výukový prostředek na Katedře kybernetiky a biomedicínského inženýrství.

Použitá literatura

- [1] MICRORISC. *Bezdrátová technologie* [online]. [cit. 2017-12-20]. 2017. Dostupné z: <https://microrisc.com/>
- [2] IQRF Technology. *About IQRF* [online]. [cit. 2018-12-21]. 2017. Dostupné z: <https://iqrf.org/>
- [3] IQRF Technology. *IQRF Low power leaflet* [online]. 2017 [cit. 2018-01-01]. Dostupné z: <https://iqrf.org>
- [4] IQRF Technology. *IQRF DPA Framework Technical Guide v3.02* [online]. 2017 [cit. 2017-12-23]. Dostupné z: <https://iqrf.org>
- [5] REX Controls. *Systém REX* [online]. 2017 [cit. 2017-12-26]. Dostupné z: <https://www.rexcontrols.cz>
- [6] RICHARDSON, Matt a WALLACE, Shawn. *Getting Started with Raspberry Pi* [online]. Vydání č. 2. USA: Makermedia, 2013 [cit. 2018-01-02]. ISBN 978-1-449-34421-4.
- [7] Raspberry Pi. *Raspberry Pi 3 Model B* [online]. 2017 [cit. 2018-01-02]. Dostupné z: <https://www.raspberrypi.org/>
- [8] NOSKIEVIČ, Petr. *Modelování a identifikace systémů*. 1. vyd. Ostrava: MONTANEX a.s., 1999. 276 stran, ISBN 80-7225-030-2.

Seznam příloh

Příloha A: Manuál pro zprovoznění řešení na prezentačním panelu I

Součástí BP/DP je CD/DVD.

Adresářová struktura přiloženého CD/DVD:

- A_DPS - Schéma, DPS a osazovací plán hardwarové části
- B_IQRF - Programy pro bezdrátovou komunikaci IQRF
- C_REX - Programy pro řídicí část v systému REX
- D_Identifikace - Experimentální identifikace soustavy
- E_Obrazová_dokumentace - Obrázky a fotografie řešení
- F_Stojan - Podkladová plocha na prezentační stojan

Postup pro spuštění aplikace

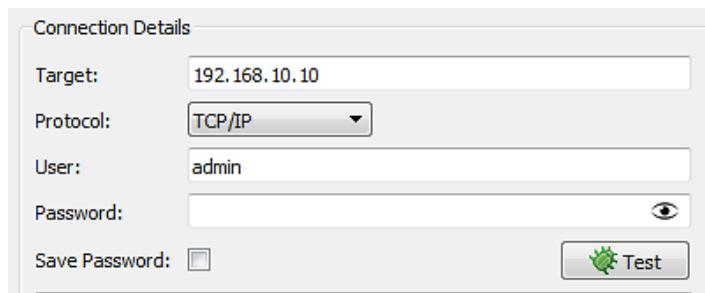
1. Zasunout jumpery na všech uzlech do pozice ON

2. Přivést napájení k desce Raspberry Pi 3

3. Spustit řídicí jádro RexCore

a) Demo verze systému REX

- Propojit PC a Raspberry Pi 3 přes Ethernet nebo připojit obě zařízení do stejné sítě
- Otevřít vytvořený soubor *serial_comm_exec.mdl* v RexDraw a stisknout „nahrát do zařízení“
- Zadat detaily připojení (viz. obrázek)



The image shows a 'Connection Details' window with the following fields and controls:

- Target:** 192.168.10.10
- Protocol:** TCP/IP (dropdown menu)
- User:** admin
- Password:** (empty text field with an eye icon for toggling visibility)
- Save Password:** ☐
- Test:** A button with a green gear icon.

- Nahrát projekt a spustit jádro RexCore

b) Plná verze systému REX

- Je-li aplikace již nahrána v cílovém zařízení, při zapnutí se jádro RexCore spustí samo, jinak je postup stejný jako u demoverze systému REX + je doporučeno zaškrtnout políčko *Store configuration permanently* pro automatické spouštění

4. Spustit vizualizaci

a) Na obrazovce cílového zařízení

- Otevřít Chromium
- Zadat adresu localhost:8008
- Vyplnit uživatelské jméno a heslo (ve výchozím režimu jméno: **admin**, bez hesla)

b) Z jiného zařízení

- Ujistit se, zda jsou obě zařízení připojeny k jedné síti, případně se připojit k platformě Raspberry Pi 3 přes Wifi Access Point
- Zadat do prohlížeče odkaz: 192.168.10.10:8008
- Vyplnit uživatelské jméno a heslo (ve výchozím režimu jméno: **admin**, bez hesla)